A Project report on

“Movie Hub”

Submitted in partial fulfilment for the requirements of the award of the degree of
BACHELOR OF COMPUTER APPLICATION

Sirifort Institute of Management Studies

Institutional Area, Sector – 25, New Delhi – 110085

AFFILIATED TO

GURU GOBIND SINGH INDRAPRASTHA UNIVERSITY

Submitted by:

Vikas Kumar(03216702017)

Guide:

Mr.Rajesh Jain

(HOD ,BCA dept)

CERTIFICATE

I **Vikas Kumar(03216702017)** certify that the project report entitled "Movie Hub" is an authentic work carried out by me at **Sirifort Institute of Management Studies.** The matter embodied in this project work has not been submitted earlier for the award of any degree or diploma to the best of our knowledge and belief.

Signature of Student:

Date:

Certified that the project report entitled "Movie Hub" done by above students is completed under my guidance.

Signature of the Guide:

Date :

Name of the Guide: Mr. Rajesh Jain

Designation: Head of BCA Department

ACKNOWLEDGEMENT

I express my sincere gratitude to my project guide **Mr. Rajesh Jain** for giving me the opportunity to prepare the project under his guidance and supervision.

Thanks to all the faculty of SIMS, Rohini for their support and criticism which enabled us to complete our project successfully.

LIST OF TABLES

LIST OF FIGURES

TABLE OF CONTENT

CHAPTER – 1

SRS

SOFTWARE REQUIREMENT SPECIFICATION

1. Introduction

1.1 Purpose

This project is aimed to provide the customers facility to book tickets for cinema halls online, through which they can book tickets anytime, anywhere. Movies Hub is basically made for providing the customers an anytime and anywhere service for booking the seat in the cinema halls and to gather information about the movies online.

1.2 Document Conventions

This document uses the following conventions.

DB - Database

ER – Entity Relationship

1.3 Intended Audience and Reading Suggestions

1.1)
The Intended Audience of this project is general public especially movie lovers and people who spent their time online.

This project main aim is to provide movie tickets to the customers .

1.4 Project Scope

Goal

The goal of the project is to provide a way for customers to buy movie tickets online. To do this:

- Provide anytime anyplace service for the customer.
- Provide effective solutions for easily availability of movie tickets.
- Provide simpler way to choose our seat and area for movie show
- To minimize the number of staff at the ticket box.
- Try to increase profit as much as possible

Objective

Through this project we present a comprehensive solution for ticket booking in multiplexes .This powerful web application is specifically designed for theater owners,to sell tickets online.

Therefore, The main objective of Movieshub is to develop an enhanced, automated, accurate, user-friendly and less maintained way of web application.

1.5 References

- https://www.w3schools.com
- https://developer.mozilla.org/en-US/
- https://stackoverflow.com

2. Overall Description

2.1 Product Perspective

Online movie ticket booking system is a replacement for ordinary movie booking system which depends on paper work for recording booking and user's information.

2.2 Product Features

Provide a simple interface and platform to ease the process of buying as well as selling products online.

Include smooth functionality and efficiency that adds to buyer's confidence.

Appropriate space is dedicated to its Offer zone and deals of the day that fetches adequate attention of users.

Ads and promotion activities motivates shoppers to buy online.

Notifying order and delivery reports to customers.

2.3 User and Characteristics

2.3.1 Administrator/Owner

1. Admin should be able to login into the system.

2. Admin should be able to insert , update and delete movies.

3. Admin can edit the information related to movies,seats,booking,payment etc.

4. Can view amount of profit/loss in a time period.

5. Administrator can update the staff information.

6. Administrator can add /remove the staff member

2.2.2 Staff

1. Staff can login into the system

2. Can add , update and delete movies

2.2.3 Normal Users

- User can Register.
- User can Login.
- User can Check Availability.
- User can Book Ticket
- User can follow Movie hub on Social Networking.

2.4 Operating Environment

The website will run on a web server supporting java script and derby database.

CO-1: The time allotted for this project will be limited to the end of this semester.

CO-2: The language for the project will be HTML and the development environment will be the embedded HTML5

CO-3: All the HTML code for the user manual will conform to the HTML 4.0 standard.

CO-4: Programming is done in JAVA, SQL, and HTML.

2.5 Design and Implementation Constraints

- The user interface shall be implemented using any tool or software package like Java JSP, Net-Beans etc.
- MS SQL Server will be used as SQL engine and database.
- The website is running 24 hours a day.
- Users may access Movie hub from any computer that has Internet browsing capabilities and an Internet connection.
- Users must have their correct e-mail and passwords to enter into their online accounts and do actions.

2.6 User Documentation

- UD-1: The system will provide an online user manual in HTML that describes the functionality and options available to the user.
- UD-2: The system will provide an online user manual in HTML that describes the policies of the website.

Some of the design and implementation constraint identified are listed below:

i. Staff members cannot edit their details.

ii. Customer would not get any details about the system.

iii. Staff members are not allowed to register, they are registered by Administrator.

2.7 Assumptions and Dependencies

Provider: We have assumed that the Movie Hub will be running on a properly working web server and database system with an Internet connection that allows this system to perform all communications with clients.

Client: We have assumed that all of the computer systems are in proper working condition and that the user is capable of operating these system's basic functions including but not limited to being 53t8able to power on the system, login and open either Internet Explorer or Mozilla Firefox, and navigate the browser to the address of this Sport Mania website.

Assumptions:

• We assume that website will open on system which support HTML and CSS. E.g. Google chrome and Mozilla Firefox etc.

• We assume that system have Internet access.

• We assume that User know how to reach the website.

• We assume that User have general idea how to move from webpage to webpage using the system.

Dependency:

Customer data being used for setup and customer recognition is dependent on information in database administered outside of the capabilities of the payment and product management.

3. System Features

Using the actor list, we begin the analysis of the client-server features that's take place in section and then for each tool that is being reported so far in section B we analysis its requirements basis in the data base editor program. All these features require that the server is on.

3.1 LOG IN**-** The users will be able to sign in to the system and can use of the specified page by accessing the website.

3.2 FORGOT PASSWORD- The users who don't remember their password can get a chance to get the new password by the help of OTP on their registered email address .

3.3 SEARCHING- The users can search movie and theatre/multiplex and can get information about their search .

3.4 SIGN UP- The users can create account for the first time by signing up and give personal details and the registration will be confirm.

3.5 BOOKING – The users can book tickets for the movies.

4. External Interface Requirements

4.1 User Interfaces

UI-1: Login and Signup Page.

UI-2: The website shall provide details of latest movies.

UI-3: These movie details can be clicked with a mouse in order to view a particular movie .

UI-4: All modifications to the database will be done through a keyboard.

UI-5: Application will be accessed through a Browser Interface. The interface would be viewed best using 1920x1080 pixels resolution setting.

UI-6: The website will provide a page for booking of ticket and for its payment.

UI-7: After all transaction the system makes the selling report as portable document format (pdf)

And sent to the customer E-mail address.

4.2 Hardware Interfaces

The System must run over the internet, all the hardware shall require to connect internet will be hardware interface for the system. The System require Database also for the store of any transaction of the system like Derby etc. System also require DNS for the naming on the internet. At the last user need web browser for interacting with the system.

4.3 Software Interfaces

The system is on server so it requires any scripting language like JavaScript etc.

The system require Database also for storing any transaction of the system like Derby etc. . System also require DNS for the naming on the internet. At the last, user needs web browser for interacting with the system.

4.4 Communications Interfaces

The Movie Hub shall use the HTTPS protocol for communication over the Internet and for the Intranet communication will be through TCP/IP protocol suite. The user must have SSL certificate licensing registered web browser.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

- You will be sign in with in 40 sec.
- If anybody make new account then he will receive confirmation email within 8 minutes.
- Search results shown within 20 sec.
- Web support 250 customers logged at the same time.

5.2 Safety Requirements

- We will try the best to make product safe.
- System would be protected by a password.
- There is no risk of mismanagement of data.

5.3 Security Requirements

- The application assumes that the user or customer is allowed to access his/her account by Email ID and password necessary to verify the identity.
- Modifications for the database shall be synchronized by the administrator.
- The system shall not leave any cookies on the customer's computer containing the user's password or confidential information.

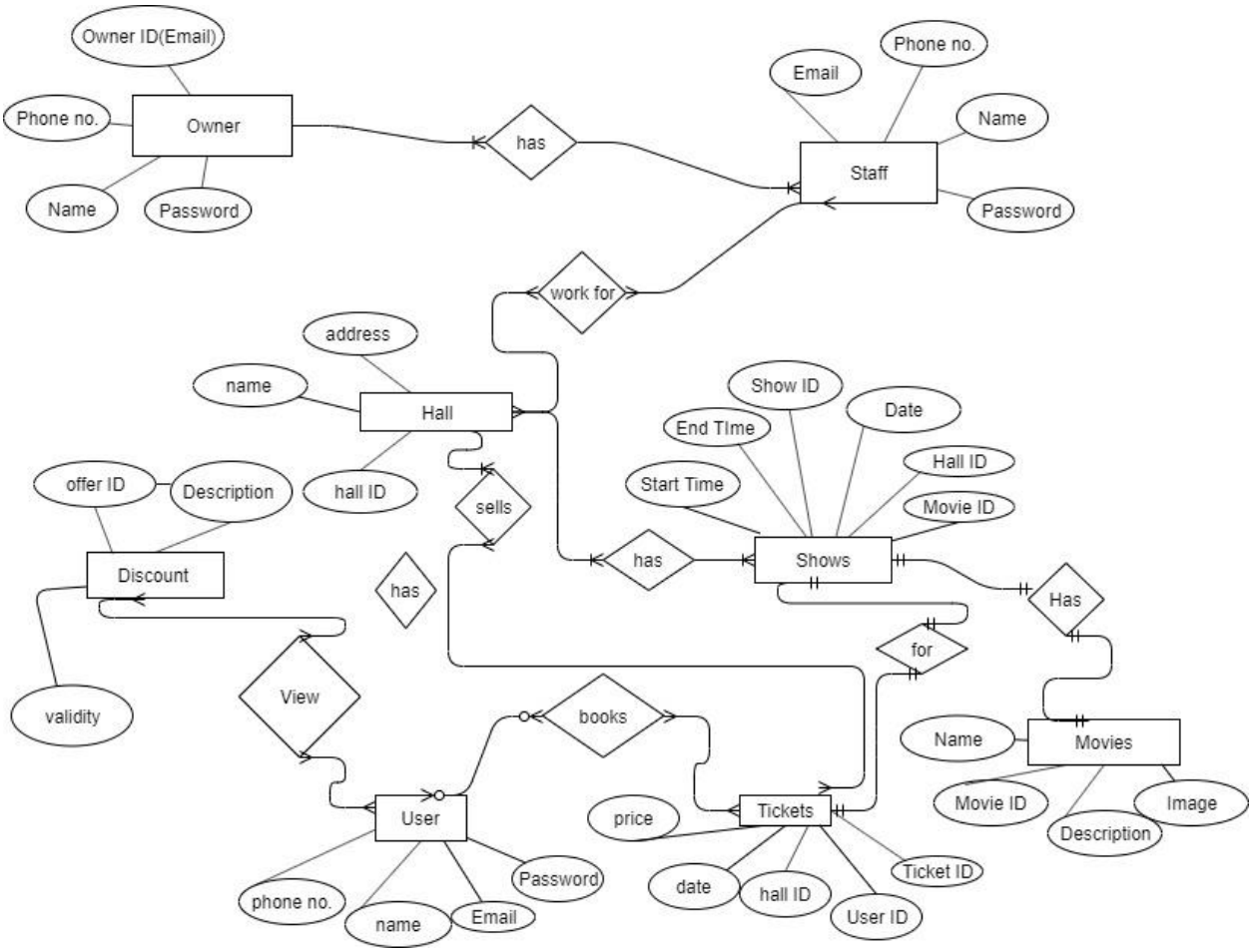
5.4 Software Quality Attributes

- **Reliability**- The system provides storage of all databases on redundant computers with automatic switchover. The reliability of overall program depends on the reliability of the separate components.
- **Maintainability**- The software design is being done with modularity in mind so that the maintainability can be done efficiently. In case of failure, a reinitialization of the program will be done.
- **Reusability** – The coding part can be use anywhere, anytime and produces simple and independent core modules that can be reused.
- **Availability**- The system shall be available all the time with enormous speed and better utilization of the time.

6. Other Requirements

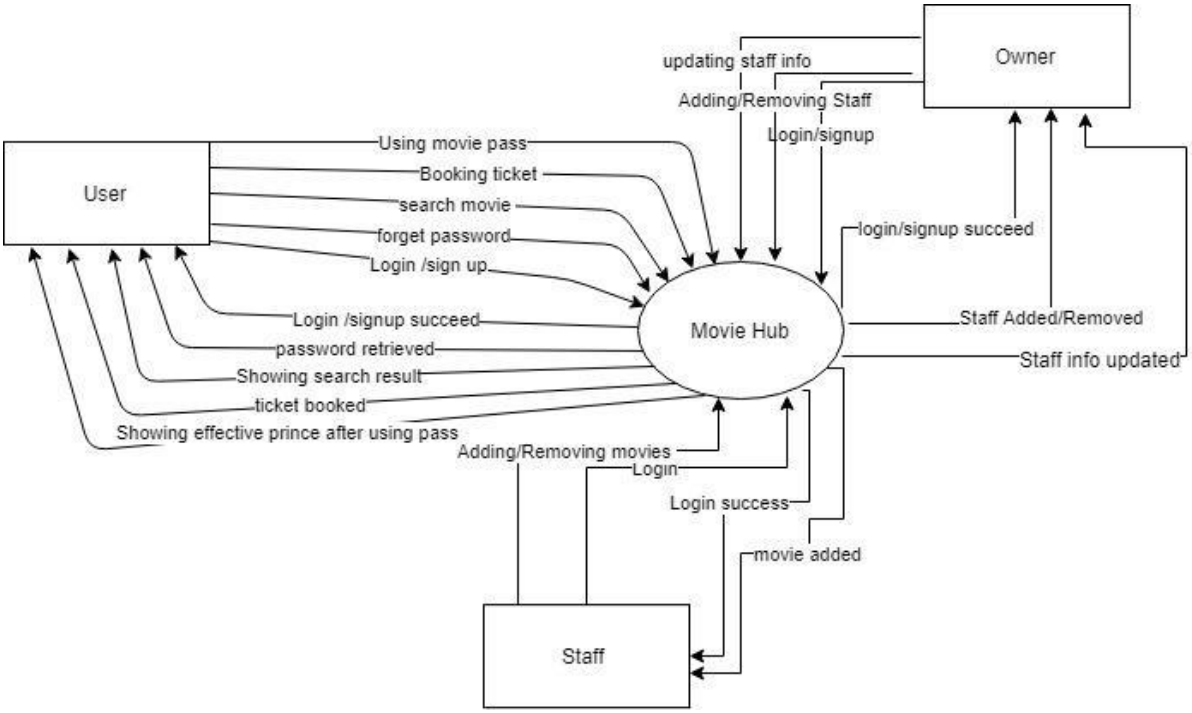
Appendix : Analysis Models

Entity-relationship diagram

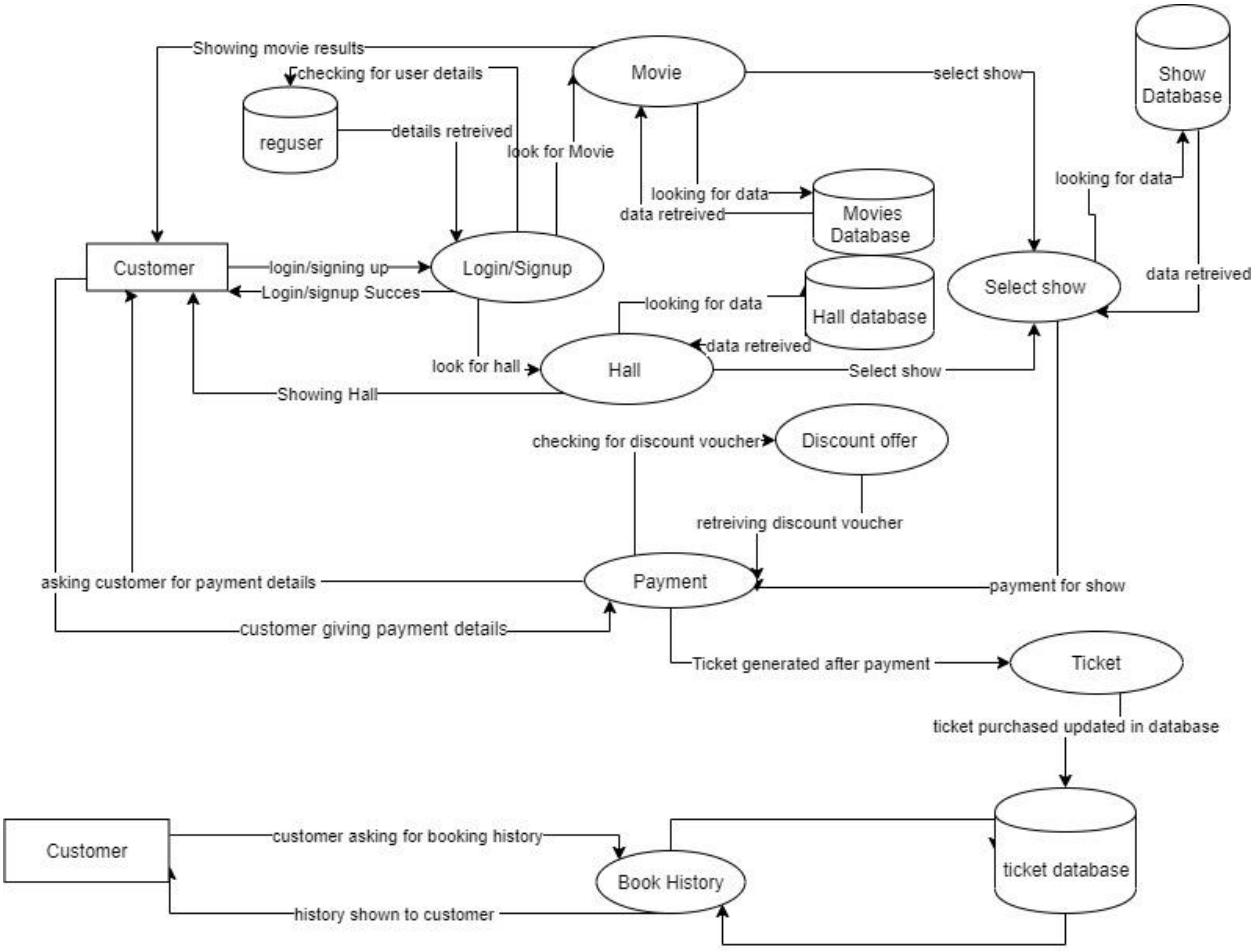


DATA FLOW DIAGRAM

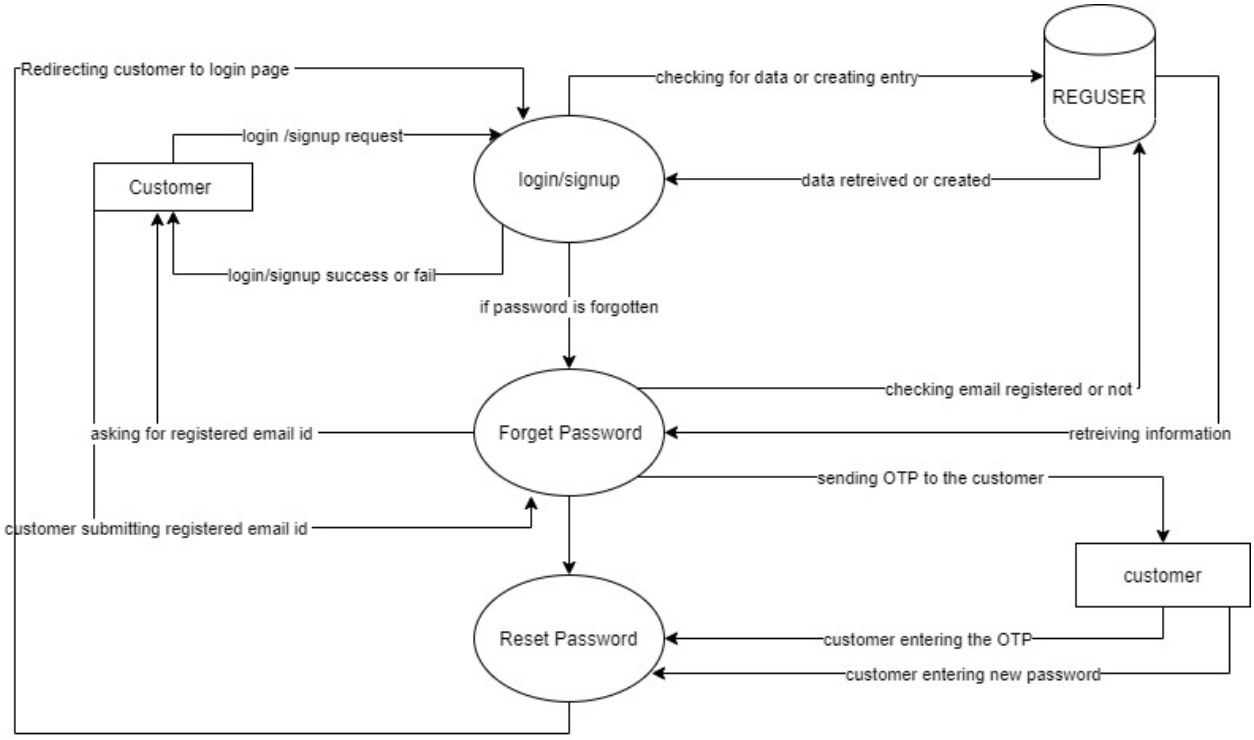
Level-0



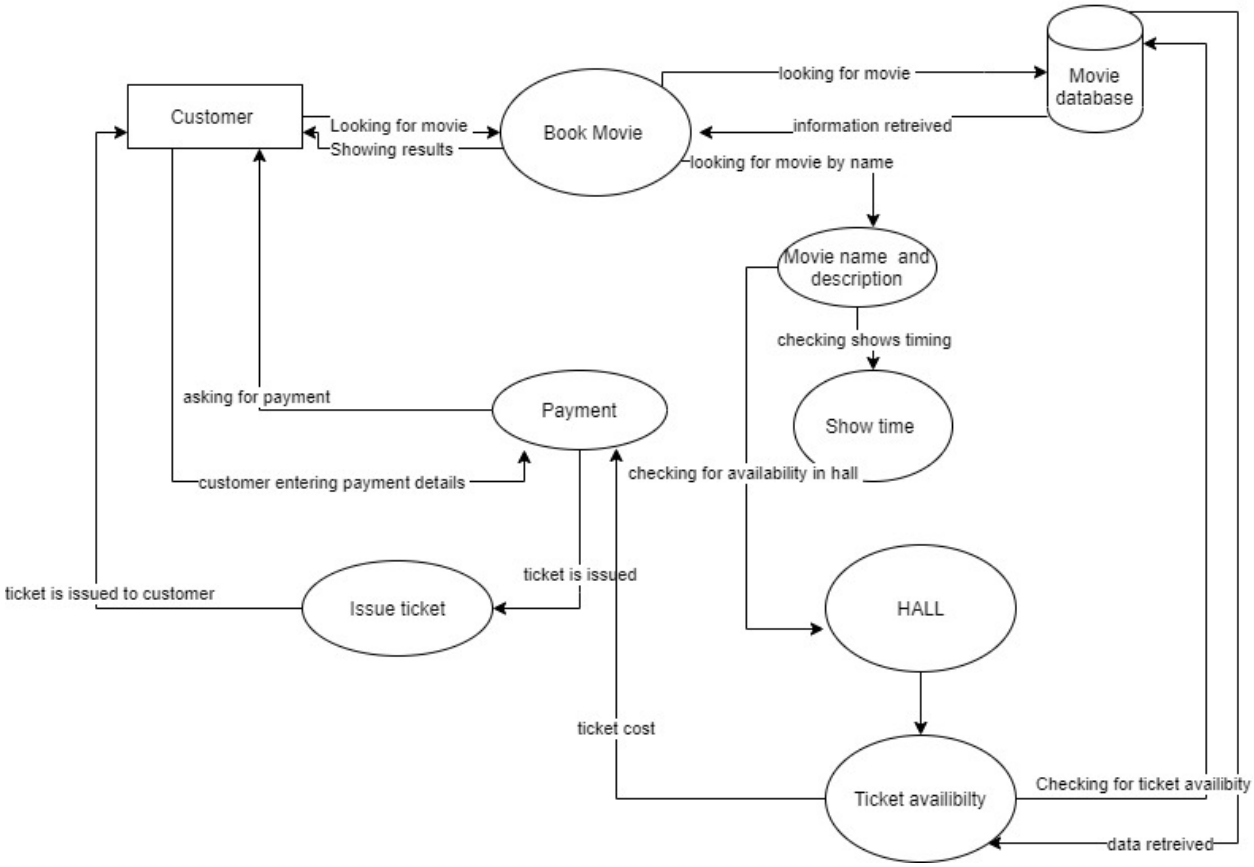
DFD level 1


Showing movie results
checking for user details
Movie
select show
Show Database
reguser
details retreived
look for Movie
looking for data
data retreived
Movies Database
looking for data
Customer
login/signing up
Login/signup Succes
Login/Signup
Select show
data retreived
looking for data
Hall database
data retreived
look for hall
Hall
Select show
Showing Hall
checking for discount voucher
Discount offer
retreiving discount voucher
asking customer for payment details
Payment
payment for show
customer giving payment details
Ticket generated after payment
Ticket
ticket purchased updated in database
Customer
customer asking for booking history
Book History
ticket database
history shown to customer

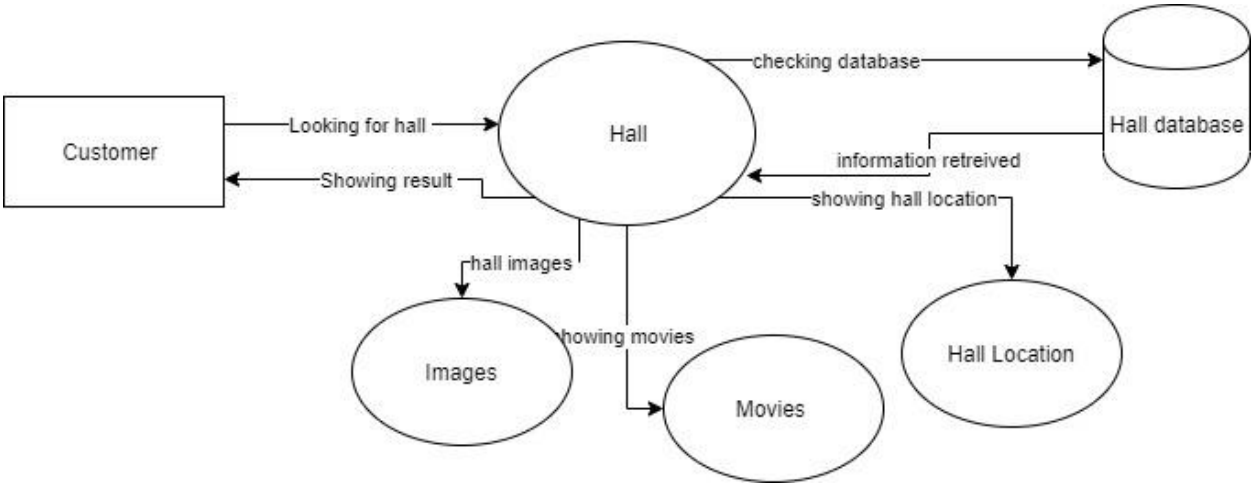
DFD LEVEL 2(Login)



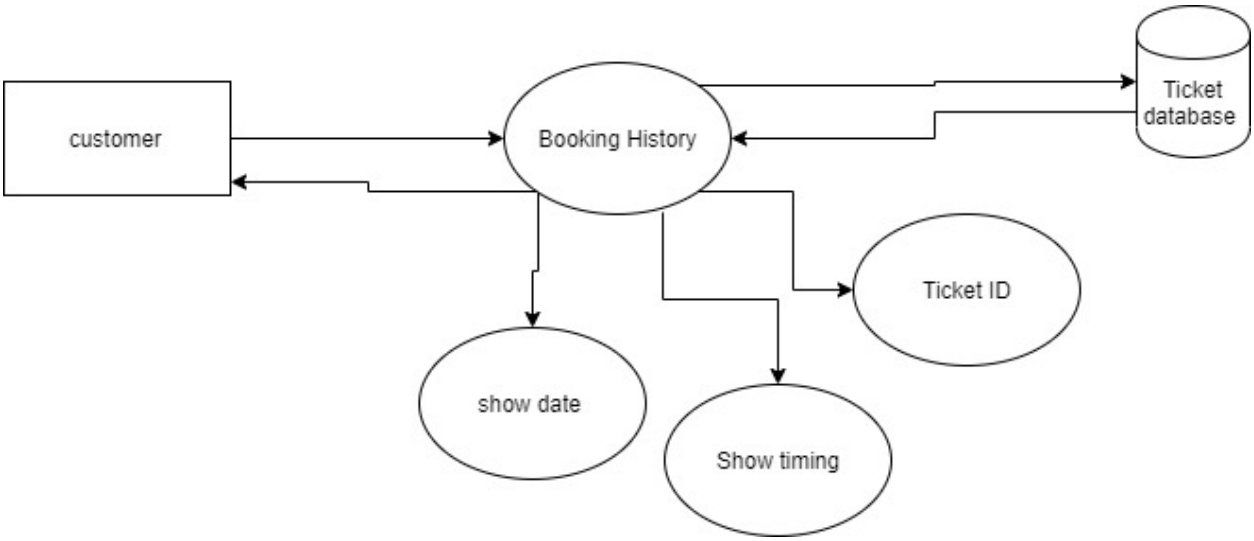
DFD LEVEL 2(Book Movie)



DFD LEVEL 2 (HALL)



DFD LEVEL 2(Booking history)



CHAPTER – 2

FEASIBILITY REPORT

1.0 GENERAL INFORMATION

1.1 Purpose

The aim of this project is to develop the website with the following objectives: -

- We create Movie hub website, through which customers can book movie tickets from anywhere and anytime in registered movie halls
- Promote efficient and effective record maintenance system that still requires physical presence of the admin and data entry operator.

1.2 Scope

The scope of this project is very high but initially the project is created for the summer training. The project purpose is to provide movie tickets to the customers and vendor can tie up with the big cinema organization which open up .

1.3 System Overview

"Movie Hub" is a website through which customer can book tickets. There are 2 entities controlling all the processes that are 1. Admin 2.Staff .
Admin can add/remove staff members and can change staff details and admin can also perform the functions of staff like adding/removing movies ,updating records. Staff can add/remove movie ,update hall details and can provide customer support. Store large database that include database for

1. Customer Details
2. Movie Details
3. Hall Details
4. Staff Details

There are three types of user i.e. Admin,Staff(DEO),Customer.

- Responsible organization: Surjan Singh Risal Singh Memorial Education Society
- System name or title: INVENTORY MODULE
- System category:

- General support system: provides general ADP or network support for a variety of users and applications
- Operational status: Under development
- System environment: Windows operating System

1.4 Project References

- www.1000projects.org
- www.google.com
- www.stackoverflow.
- Documentation related to the project
- Ideas in mind about the project

1.5 Acronyms and Abbreviations

IP	Information Protection
SICM	Standard Input Configuration Management
SOCM	Standard Output Configuration Management
XHTML	Extensible Hypertext Markup Language
SCA	Standard Cabinet Assembly

1.6 Points of Contact

1.6.1 Information

After thoroughly analyzing the existing system the following objectives were set:-

- Providing user friendly interface
- Interaction without any interruption

- User should be registered in the database of the college
- User should have login id and password
- Both the users must be different
- Keeping project records in the software
- Maintaining data consistency
- Adequate validation checks for data entry

1.6.2 Coordination

S.No	Name	E.no	Designation	Email ID
1	Vikas Kumar	0316702017	Analyst & programmer	kuvikasar112@gmail.com
2	Rishabh Rawat	02516702017	Testing	rishabhrawat2323@gmail.com
3	Abhishek Rawat	40216702017	Database Admin	Abhishek.rwt2076@gmail.com
4	Gaurav Singh Thapa	40616702017	Documentation	G.thapa9248@gmail.com

Note: - As this is the training project created at college level as per our syllabus. So there is no interaction needed with any other organization.

2.0 MANAGEMENT SUMMARY

2.1 Environment

2.1.1 Organizations Involved

Responsible organization: Surjan Singh Risal Singh Memorial Education Society
Sponsor: Sirifort Institute of Management Studies
Developer: Vikas Kumar, Rishabh Rawat, Abhishek Rawat,Gaurav Singh Thapa
User: Admin, Staff, Customer

This project will be a website application to be developed in Java using JDK.

- NetBeans IDE 8.0.2

2.1.2 Input/ Output

The system requires following input and output: -

INPUT REQUIRMENTS

- Name
- Email Address
- Password

OUTPUT REQUIRMENTS: -

- Mails
- Message
- Notices

2.1.3 Processing

HARDWARE REQUIRMENTS: -

Development Time

1. Memory minimum 32 GB (Approx.)
2. Other standard cabinet assembly

Deployment Time

1. Memory minimum 10 GB (Approx.)
2. Other standard cabinet assembly

DOCUMENTATION AND PRESENTATION TOOLS:-

1. Microsoft Word (2013)

SOFTWARE REQUIRMENTS

Development Time

1. Operating System (windows 7)
2. Browser
3. NetBeans IDE 8.0.2
4. Microsoft Office (2003)

Deployment Time

1. Operating System (XP)
2. Browser

2.1.4 Security

This program uses object oriented mechanisms to protect its data passed using methods. Also at the entry level Email ID and password is required to enter into the system. After that a session is maintained at each level of module.

2.1.5 System Interaction

This is movie booking website and the project is carried on by our each member of the group each member of the group has contributed in the project. The customer can interact with the website very easily and the can book the tickets . Customer email id and password is neccasry to register in the website. Admin can look into the staff and and user information. Staff can add/remove movies and change hall details etc. Some of the major modules are
: Hall, Movie, Offers .

2.1.6 Physical Environment

Since we will be creating this module at the college level so we are not focusing on any particular organisation we took some cinema halls at random to explain the working of the project. The project is organistion independent

2.2 Current Functional Procedures

- The current functional procedures of the system are purely manual as the system cannot perform any task of its own. User has to give the command to perform any specific task then only system will produce output.

- The cost incurred in operating the current system is considerable as it is a project of a high-level language.
- The people involved in the project should be skilled up to the level at which they are familiar with the JAVA Programming, DFD, ER Diagrams, NetBeans.

- The current system requires only few members of staff, as the maintenance is not so much required.

- Excellent attention to details.

- Proven ability to work as a team members.

2.3 Functional Objectives

Analyze the anticipated functions of the system, considering such areas as new services, increased capacity, legislative and policy requirements, privacy and security requirements, audit controls, and target completion date.

The following objective must be taken into account while developing Movie Hub (Movie ticket booking website).

Speed - Speed is defined as the time taken to implement the Movie hub. While designing movie hub the time taken by the organization to implement Movie hub must be taken into account.

Scope – Movie hub system should consider and include all the functional and technical characteristics required.

Resources – Resources are everything that is needed to support the project. This includes people, hardware, system, software, etc.

Risk - Risk is defined as the factor that resists the overall success of Movie hub implementation. Therefore, all the risk that will arise must be taken into account while developing Movie hub.

Complexity – Complexity is the degree of difficulty anticipated during implementation, operation and maintaining the Movie hub systems.

Benefits – To get the maximum benefit from Movie hub, care must be taken to design the Movie hub by following the procedures followed by the organization.

2.4 Performance Objectives

All the software and the products used to develop the project are used efficiently and the project that is being developed by us is managed properly.

Major Performance Objectives include:-

Processing speed: as manual task is reduced, processing speed of work will be increased.
Services:-management information services will be improved due to ease to get information from database.

Decision making system: - control over decision making system will be improved due to simple click and get method. This click and get method is required to reduce the effort of getting the information from database.

2.5 Assumptions and Constraints

Different assumptions and constraints are taken in consideration while preparing this system such as:

Operation life:-This is the training project made by BCA students at college level. It is operational till the end of the training.

Period of time for comparison of system alternatives:-Before deciding the different modules of the project, around two months devoted in the research of the modules provided by the different other alternative system.

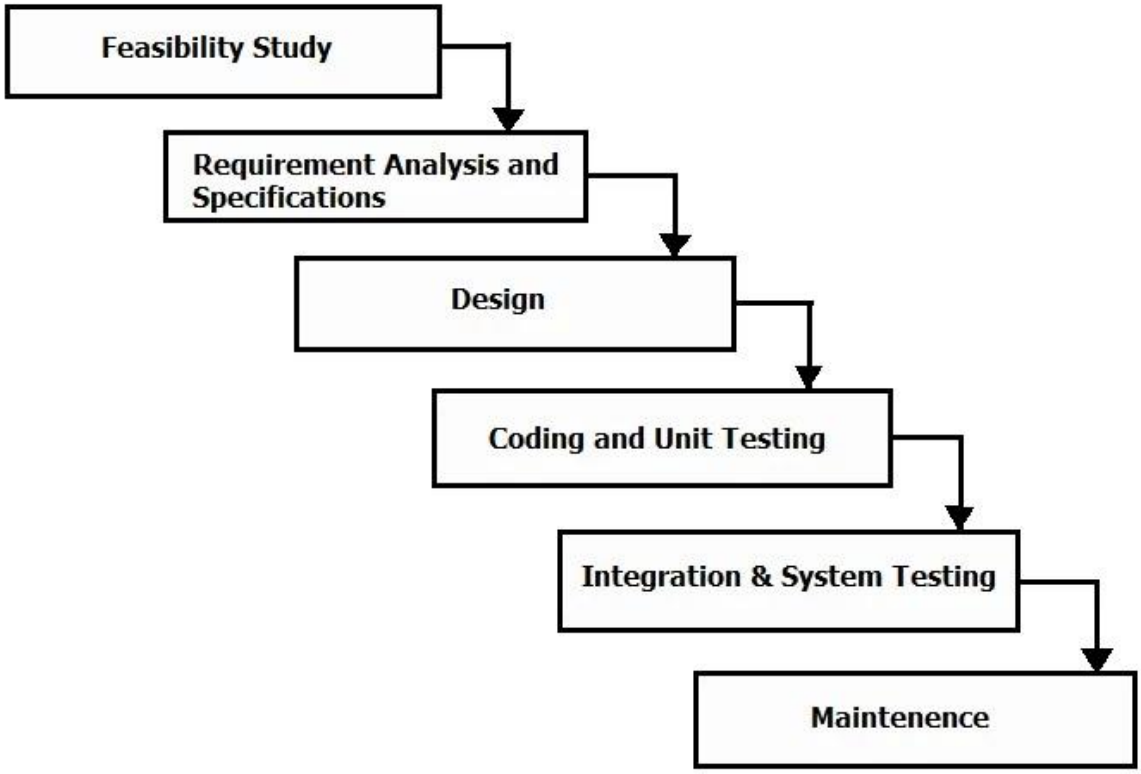
Operation environment:-It is assumed that it will perfectly run ion any operating environment.

Availability of information:-It is assumed that all the information that are required for this project is available while developing the logics of the project.

2.6 Methodology

We studied on the internet regarding the project content to make it effective and use by a simple person. We have also various projects to make our project worth of doing all the tasks, which are not done by other projects.

WATERFALL MODEL



2.7 Evaluation Criteria

A responsibility of this application is to provide access to view the demand and availability of the inventories so that the work is efficient and there will be no wastage of time and memory.

Priority: Priority is given due to following reasons
Easy Access to Database.
Reduce man power resources.
User friendly approach.
Reduced paper work.
Fast and easy to use
Security

2.8 Recommendation

We take into account the suggestion from the participation of the survey and try to implement all the features that we can put into the application.

3.0 PROPOSED SYSTEM

3.1 Description of Proposed System

This system will be used in three user modules which are ADMIN, STAFF, CUSTOMER.

As all of these have different requirements, the modules are designed to meet their need and avoid any type of confusion. The uses of all two user modules are described below: -

1. ADMIN

- Login
- View staff and customer details
- View static information
- Querying
- Can perform all the tasks of Staff

6.1 2. STAFF

- Login
- Add user details
- Add new modules
- View static information
- Add movie
- Update details

3. CUSTOMER

- Register
- Login
- Search Movies /Hall
- Book Tickets
-

3.2 Improvements

Improvements are the modification that is regularly needed in any system at the maintenance time to improve its action.

The improvements of the system in terms of the objectives are as follows:-

- The project will be more feasible.
- An application that tells the timing and have records of student login for protection of their account.

3.3 Time and Resource Costs

As this is the project made at college level so there is no much funding require in monetary terms but need to utilize more time in collection of data through the means of survey

conducted at the different organizations and societies through distributing on paper forms and online Google forms.

Also more time is utilize in collecting information and resources required in developing the project.

This is the project made by a group at college level so there is no staff needed to be recruited for different purposes.

We have invested 5 MONTHS to collect the information needed for development of the project.

3.4 Impacts

3.4.1 Equipment Impacts

Updated and currently launched hardware are required for perfectly running this project.

We are using NetBeans IDE 8.0.2 and JDK (database); no other equipment requirement is there

3.4.2 Software Impacts

To run this system accurately, it is preferred to use browser like Google Chrome, Mozilla Firefox, etc.
NetBeans IDE 8.0.2 for the development of system.

3.4.3 Organizational Impacts

In order to maintain coordination between the different modules of the project the staff needs to have some technical knowledge about the organization or the basic system knowledge.

3.4.4 Operational Impacts

Effects of different operations on the system, as:

* User operating procedures: User needs to have some knowledge about the procedure to communicate with system. It will be improved due to user friendly interface.

* Operating center procedures: Different operating centers (staff) have different facilities to operate with the system. The project will benefit the management and developers, as manual work and efforts will be reduced.

* Operating center/user relationships: The operating center\user relationships will improve if we are provided some information about the project from the college campus.

* Source data processing: This will depends on the needs in project completion.

* Data entry procedures: Forms having buttons, text-boxes, checkbox, data grid view etc. All the details of various users, students, companies, books etc needs to be entered into the database.

* Data retention requirements: These are provided itself in the database, information storage and retrieval procedures are provided in the database. Database will be non-redundant.

* Output reporting procedures, media, and schedules: Media involved are the machine systems. Schedules are user defined.

* System failure contingencies and recovery procedures: Backup of the database is needed in case of emergency or system failure.

3.4.5 Developmental Impacts

Describe the developmental impacts, such as:

- Specific activities to be performed by the user in support of development of the system
- Resources required to develop databases
- Computer processing resources required to develop and test the new system
- Privacy and security implications

3.4.6 Site or Facility Impacts

All the project work is done in the college campus.

3.4.7 Security and Privacy Impacts

Information Security:-The security of information is very important. Information can be of items, prices or any one must be secure.

Physical Security:-Destruction/damage of any asset of item may cause fine.

Personnel Security:-Disclosure, modification, interruption, disruption, removal of personal information by disgruntled.

Information Protection (IP):-The information of USER must be secure so no one can misuse the information.

We have installed Password on the system to protect the loss or theft of data.

3.5 Rationale for Recommendations

There are many alternative systems that are already in work. After judging those systems, we have made many improvements in our system such as admin update the detail of the inventory.

CHAPTER – 3

SDS

SOFTWARE DESIGN SPECIFICATION

1. Revision History

Version	Name	Reason For Changes	Date

2. Approved By

Approvals should be obtained from faculty/ HOD

Name	Signature	Department	Date

3. Introduction

This document is the design report for a web-based Movie ticket booking system. This is mainly about 'how to do' and also will help provide an insight to the whole system design and implementation of the online movie booking system. This software has the following three main components:

It is my desire that a completed software design specification meet the following criteria:

1. Implement the different types of user – Administrator, Customer and Staff
2. Implement cart for checkout and payment procedures.
3. Management of movie, shows and tickets. This design document mainly consists of State Machine/ Activity Flow Diagrams, Class Design, Internal Data Structures, Architectural design, User Interface and Testing. The main purposes of this design document are listed below.

1. Precise understanding of the requirements and constraints related with the programming language, and User Interface.

2. System decomposition into manageable units or modules

3. Abstraction of the system implementation with the help of classes

4. Provide a basic outline of the User Interface of the online shopping mall.

Please note that there are no sections in this document for describing administrative or business duties, or for proposing plans or schedules for testing or development. The sections in this document are concerned solely with the design of the software. While there are places in this document where it is appropriate to discuss the effects of such plans on the software design, it is this author's opinion that most of the details concerning such plans belong in one or more separate documents.

3.1 Document Outline

Here is the outline of the proposed template for software design specifications.

- Introduction

- System Overview
- Design Considerations
    - Assumptions and Dependencies
    - General Constraints
    - Goals and Guidelines
    - Development Methods

- Architectural Strategies
    - Login
    - Sign Up
    - Home Page
    - Payment Page
    - Contact Page
    - About Page
    - Profile Page
- System Architecture
    - DFD-0
    - DFD-1
    - DFD-2
    - ER DIAGRAM
- Policies and Tactics
    - Requirement of specific product
    - Program idioms
    - Plans for testing the software
    - Interfaces for end users, software, hardware and communication
    - Compilation
- Detailed System Design
    - Classification
    - Definition
    - Responsibilities
    - Constraints
    - Compositions

- Database Design

- Glossary
- Bibliography

The above outline is by no means exclusive. A particular numbering scheme is not necessarily required and you are more than welcome to add your own sections or subsections where you feel they are appropriate. In particular you may wish to move the bibliography and glossary to the beginning of the document instead of placing them at the end.

The same template is intended to be used for both high-level design and low-level design. The design document used for high-level design is a "living document" in that it gradually evolves to include low-level design details (although perhaps the "Detailed Design" section may not yet be appropriate at the high-level design phase).

The ordering of the sections in this document attempts to correspond to the order in which issues are addressed and in which decisions are made during the actual design process. Of course it is understood that software designs frequently (and often fortunately) don't always proceed in this order (or in any linear or even predictable order). However, it is useful for the purpose of comprehending the design of the system to present them as if they did. Frequently, one of the best ways to document a project's design is to keep a detailed project journal, log, or diary of issues as they are mulled over and bandied about and to record the decisions made (and the reasons why) in the journal. Unfortunately, the journal format is not usually organized the way most people would like it for a formal review. In such cases, for the purpose of review, the journal can be condensed and/or portions of it extracted and reorganized according to this outline. However, if this is done then you need to choose whether to update and maintain the design document in the journal format, or the formal review format. It is not advisable to try and maintain the design document in both formats. (If you have an automated method of converting the journal into a formal document, then this problem is solved.)

3.2 Document Description

Here is the description of the contents (by section and subsection) of the proposed template for software design specifications:

3.2.1 Introduction
Provide an overview of the entire document:

This document will cover all the necessary detailed related to the designing specifications of the software.

Introduction

This movie ticket booking website helps people to buy movie tickets from anywhere with the help of internet accessing device.

Store database that include database for registered visitors and signed in visitors and also for booking the car.

The user here is a visitor:

1. Movie
2. Hall
3. Bookings
4. Feedback
5. Offers

1. Purpose

The main goal of preparing this document is to describe the decisions taken in a process of creating this group project called as MOVIE HUB.

2. Scope

Scope of the project is to specify the entire group project called MOVIE HUB

3. The system/product using any applicable names and/or version numbers

- Operating System(windows 10)
- Browser
- Net Beans IDE 7.3.1
- Microsoft Office 2013

4. Intended Audience and Reading Suggestions

This document is intended for any individual user, tester, developer, project manager or documentation writer that needs to understand the basic system architecture and its specifications.

Developer: The document is to be utilized by the professor to evaluate the software's design and features.

Visitor: This project is for the persons who likes to view upcoming and latest or innovative module design. They can also view the different services provided by the developers of the website.

Tester: The tester needs this document to validate that the initial requirements of this programs actually corresponds to the executable program correctly. This document contains the necessary requirement and some aspects of the analysis of the requirements and is organized based on the IEEE Standard for Software Requirements Specification.

MAIN MODULES ARE:

1. User
2. Bookings
3. Movie
4. Offers
5. Payment
6. Feed Back
7. Login
8. Sign-Up

We are working on Movie Booking system.

System Overview

Our System has been designed to help the user to have a view of cars, moreover the model and specifications. User can pre-book the car. Driver information is also provided in the website. However, the entire solution is designed based on a module approach that gives flexibility to our users to choose desired modules as per their requirements.

4. Design Considerations

The project design was designed using Net Beans 7.3.1, So tools available by the following software was used and the best ability of the developing team to create the project Movie Hub. Information on working of different Movie Booking site was collected to enhance the user experience as well as their design was taken into consideration

4.1 Assumptions and Dependencies

The users can see their Bookings/tickets at any time by just entering the details.

- Related software or hardware
- Operating systems
- End-user characteristics
- Possible and/or probable changes in functionality
- Roles and regulations are already established.

Dependencies

This software is dependent on Movie Booking Module

4.2 General Constraints

- Glassfish 4.0 is the used because it support wide range of user access, at the same time Glassfish 4.0 server has design to help provides the highest level of security for enterprises data through features such as databases encryption ,more secure default setting, password policies enforcement, granular permissions control,& an enhanced security model.
- Net beans are used as a frontend using JSP.Verification and validation requirements (testing)

4.3 Goals and Guidelines

The website efficiently reduce man work and provides easy, quick and global access to the various detailed. This website is useful for the users who generally watch movies in cinema halls as the site provide tickets very conveniently. The user must have the following characteristics:-

1. User details(customer): His/hers details like Name, Phone No., Email Id, Password.

4.5 Development Methods

The establishment and use of sound engineering principles in order to obtain economically developed software that is reliable and works efficiently on real machines is called software engineering:-

SOFTWARE ENGINEERING is the discipline whose aim is:-

1. Production of quality software.
2. Software that is delivered on time.

3. Cost within the budget.
4. Satisfies all retirements.

Software process is the way in which we produce the software. A part from hiring smart, knowledgeable engineers and buying the latest development tools, effective software development process is also needed, so that engineers can systematically use the best technical and managerial practices to successfully complete their projects.

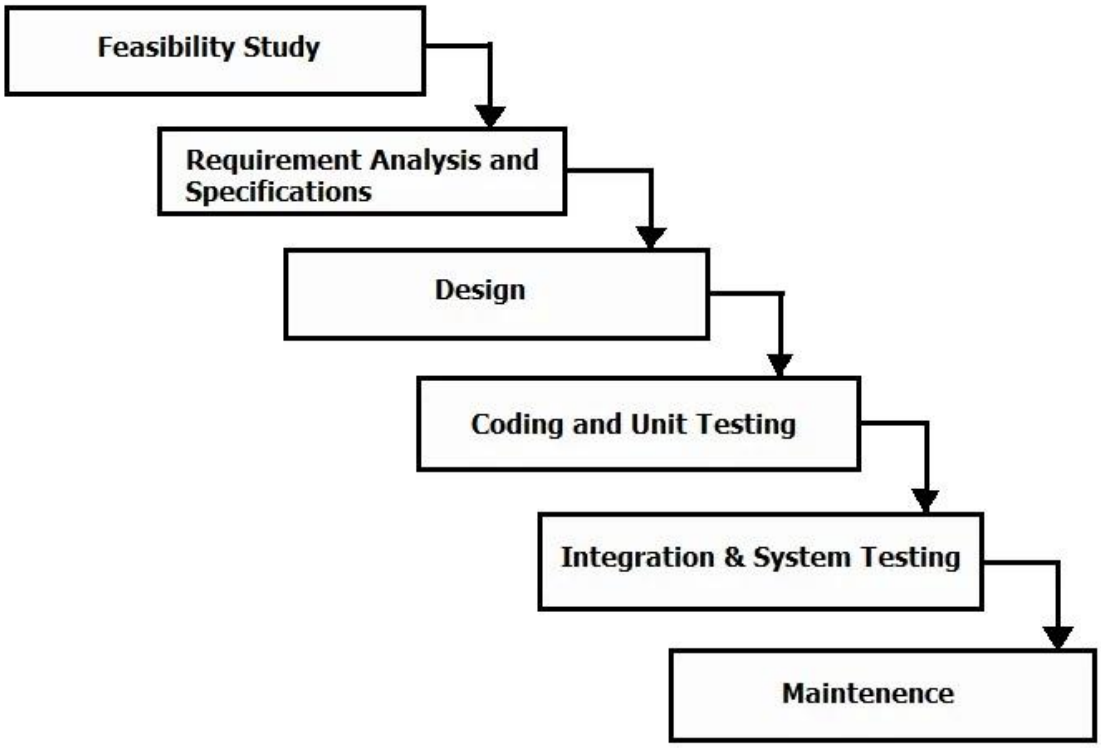
A software life cycle is the series of identifiable stages that a product undergoes during its lifetime. A software lifecycle model represents of the software life cycle. A life cycle model represents all the activities required to make a software product transmit through its lifecycle phases. It also captures the order in which these activities are to be taken.

LIFE CYCLE MODELS:

There are various life cycle models to improve the software processes.

- WATERFALL MODEL
- PROTOTYPE MODEL
- ITERATIVEENHANCEMENT MODEL
- EVOTUTIONAR MODEL
- SPIRAL MODEL

In this project, Waterfall model is followed.


Feasibility Study
Requirement Analysis and Specifications
Design
Coding and Unit Testing
Integration & System Testing
Maintenence

WATERFALL MODEL

This model contains 6 phases:

- **Feasibility study:** The feasibility study activity involves the analysis of the problem and collection of the relevant information relating to the product. The main aim of the feasibility study is to determine whether it would be financially and technically feasible o develop the product.
- **Requirement analysis and specification**

  The goal of this phase is to understand the exact requirements of the employees and students and to document them properly.
- **Design**

  The goal of this phase is to transform the requirement specification to a structure that is suitable for implementation in some programming language.
- **Implementation and unit testing**

  During this phase the design is implemented. Initially small modules are tested in isolation from rest of the software product.
- **Integration and system testing**

  In this all the modules are integrated and the tested altogether.
- **Operation and maintenance**

  Release of software inaugurates the operation and life cycle phase of the operation.

5. Architectural Strategies

• Module design

The system design shows how software will be structured to satisfy the requirements identified during the analysis phase. The design process is a translation of requirements into a description of the software structure, software components, interfaces and data necessary for the implementation phase. The design phase provides a complete blueprint for the implements activity.

Movie Hub

- **Home Page**
  - Customer can browse through the site and can see the movies showing/available for booking on the site.

- **SignUp**
  - Customer is required to give personal information to create account which will hold all the bookings, payments, address etc.

- **Login**
  - Customers and admin are given specific ID and Password to easily access their details and booking.

- **Movie**
  - Customer can browse on this page and can see which movies are available for booking. Customer can also watch trailer and can see the cast details in module too.

- **Hall**
  - Customer can see which cinema halls are registered with the website and this module also provide the details of the cinema hall i.e. address and contact number.
- **Offer**
  - Provide various offers.

- **Feedback**
  - Customers can give their feedback to us.

6. System Architecture

About System Architecture

DATA FLOW DIAGRAM

A DFD also known as 'bubble chart' has the purpose of clarifying system requirements and identifying major transformations. It shows the flow of data through a system. It is a graphical

tool because it represents increasing information flow and functional details. Four simple notations are used to complete a DFD. These notations are given below: -

DATA FLOW

The data flow is used to describe the movement of information from one part of the system to another part. Flows represent data in motion. It is pipe line through which information flows. Data flow is representing by an arrow.

Process

A circle or bubble represents a process that incoming data to outgoing data. Process shows a part of the system that transforms inputs to outputs.

External Entity

A square defines a source or destinations of system data. External entities represent any entity that supplies or receives information from the system but is not a part of the system.

Data store

The data store represents a logical file. A logical file can represent either a data store symbol which can represent either a data structure or physical file on disk. The data store is used to collect data at rest or temporary of data. It is represented by open rectangle.

6.1 Subsystem Architecture

If a particular component is one which merits a more detailed discussion than what was presented in the system architecture section, provide that more detailed discussion in a subsection of the system architecture section. If necessary, describe how the component was further divided into subcomponents, and the relationships and interactions between the subcomponents.

If any subcomponents are also deemed to merit further discussion, then describe them in a separate subsection of this section. Proceed to go into as many levels/subsections of discussion as needed in order for the reader to gain a high-level understanding of the entire system or subsystem.

If this component is very large and complex, you may want to consider documenting its design in a separate document and simply including a reference to it in this section. If this is the option you choose, the design document for this component should have an organizational format that is very similar this document.

7. Policies and Tactics

- Administrators are assigned with their default password and they can change it afterwards.
- For coding of different forms we have used separate files and query. A simple HTML form calls a file for insertion in database.
- For easy understanding of file operation simple names are used for naming the files.
- Every administrator need first login to go in the corresponding section.

8. Detailed System Design

The application will have a user friendly and menu based interface. It will have a login screen for entering user-name, password will be provided. Access to different screens will be based upon the user. There is a screen for displaying information regarding filling of customer details.

8.1 Classification

The kind of component, such as a subsystem, module, class, package, function, file, etc.

8.2 Definition

The specific purpose and semantic meaning of the component. This may need to refer back to the requirements specification.

8.3 Responsibilities

Car Rental module provides the following services and facility to the end users:

- It has user friendly interface having quick authenticated access of documents.
- Easily scalable to grow with changing system requirement.
- Increase the processing speed.
- Easy retrieval of information.
- Easy verification and monitoring of records.
- Quick query processing.
- Easy access to data.
- Reliable and efficient.

8.4 Constraints

Any relevant assumptions, limitations, or constraints for this component. This should include constraints on timing, storage, or component state, and might include rules for interacting with this component (encompassing preconditions, post conditions, invariants, other constraints on input or output values and local or global values, data formats and data access, synchronization, exceptions, etc.)

8.5 Composition

A description of the use and meaning of the sub components that are a part of this component.

8.6 Database Design

The database design specifies how the date of the software is going to be stored.

8.7 Table schemas

The complete (compliable) set of CREATE TABLE statements (and other SQL statements) that declare the database schema, including integrity constraints, domain specifications, assertions, and access privileges -- documented in a template with the intended use of each table and column.

This is a suggested template you may use[1]:

Table-1: User

COLUMN NAME	DATA TYPE	LENGTH	KEY CONSTRAINTS	DESCRIPTION
Name	Varchar	40		It specifies the name
Email	Varchar	40	Primary Key	It specifies the id
Password	Varchar	256		It specifies the password
Mob	numeric	11		It specifies the mobile no
Type	varchar	40		It specifies the the type of user – user/admin

SQL Code	Create table Reguser
	{
	Name varchar(40),
	Email varchar(40) Primary Key,
	Password varchar(256),
	Mob numeric(10),
	Type varchar(10)
	};			

Table-2: Tickets

COLUMN NAME	DATA TYPE	LENGTH	KEY CONSTRAINTS	DESCRIPTION
TicketID	Varchar	10	Primary Key	It specifies the Ticket id
UserID	Varchar	40	FK	It specifies the id
Seats Selected	numeric	2		It specifies the seat selected
Price	Int	10		It specifies the price
Show Date	Date			It Specifies the show date
Hall ID	varchar	10	FK	It specifies the hall ID
Show ID	numeric	10	FK	Its Specifies the Show ID
Screen ID	numeric	10	FK	It specifies the Screen ID
Movie Name	numeric	10		It specifies the movie name

SQL Code	Create table Tickets
	{
	TicketID varchar(10) Primary Key,
	UserID varchar(40) Foreign Key,
	SeatsSelected numeric(2),
	Price int(10),
	ShowDate date(8),
	HallID numeric(10) Foreign Key,
	ShowID numeric(10) Foreign Key,
	ScreenID numeric(10) Foreign Key,
	MovieName(10),
	};			

Table-3: Movie

COLUMN NAME	DATA TYPE	LENGTH	KEY CONSTRAINTS	DESCRIPTION
ID	Varchar	10	PK	It specifies the movie name
Name	Varchar	10		It specifies the name
Description	Varchar	256		It specifies the Description
Image	varchar	100		It specifies image path.
Cast	varchar	256		It specifies cast in the movie.
Trailer	varchar	200		It is the youtube link of the movie's trailer
Thumbnail_img	varchar	100		It specifies cover pictures path

SQL Code	Create table Movie
	{
	ID varchar(10) Primary Key,
	Name varchar(10),
	Description varchar(256),
	Image varchar(100),
	Cast varchar(256),
	Trailer varchar(200),
	Thumbnail_img varchar(100);
	};			

Table-4: Shows

COLUMN NAME	DATA TYPE	LENGTH	KEY CONSTRAINTS	DESCRIPTION
ShowID	varchar	10	PK	It specifies the showid
ShowDate	Date			It specifies the Show Date
Start Time	Time			It specifies the Start Time
End Time	Time			It specifies the End Time
MovieID	Varchar	10	FK	It specifies the MovieID
ShowDate	Date	10	FK	It specifies the HallID
Seats	Numeric	2	FK	
Price	Int	10		
SQL Code	Create table Shows {ShowID varchar(10) Primary Key, ShowDate Date(), StartTime(), EndTime(), Seats numeric(2), Price int(10), HallID numeric(10) Foreign Key, MovieID numeric(10) Foreign Key, };			

Table-5: Hall

COLUMN NAME	DATA TYPE	LENGTH	KEY CONSTRAINTS	DESCRIPTION
HallID	**Varchar**	**10**	**PK**	**It specifies the HallID**
Name	**Varchar**	**10**		**It specifies the Name**
Location	**Varchar**	**15**		**It specifies the hall location**
Contact_NO	**Numeric**	**10**		**It describes the contact number of hall.**
Contact_Email				
Img				
SQL Code	**Create table Hall** **{** ShowID varchar(10) Primary Key, UserID varchar(40) Foreign Key, SeatsSelected numeric(2), Price int(10), ShowDate date(8), HallID numeric(10) Foreign Key, ShowID numeric(10) Foreign Key, ScreenID numeric(10) Foreign Key, MovieName(10), **};**			

Table-6: Offer

COLUMN NAME	DATA TYPE	LENGTH	KEY CONSTRAINTS	DESCRIPTION
OfferID	Varachar	10	PK	It specifies the OfferID
Description	Varchar	100		It specifies the offer description
Validity	date	8		It specifies the offer validity(date)

SQL Code	Create table DiscountOffer
	{
	
	offerID varchar(10) primary key,
	Description varchar(100),
	Validity date,
	
	};			

Tables' data:

The tables have to be populated by you and your client. Each table must contain an appropriate number of data. The contents of the tables have to be provided (in an organized way.)

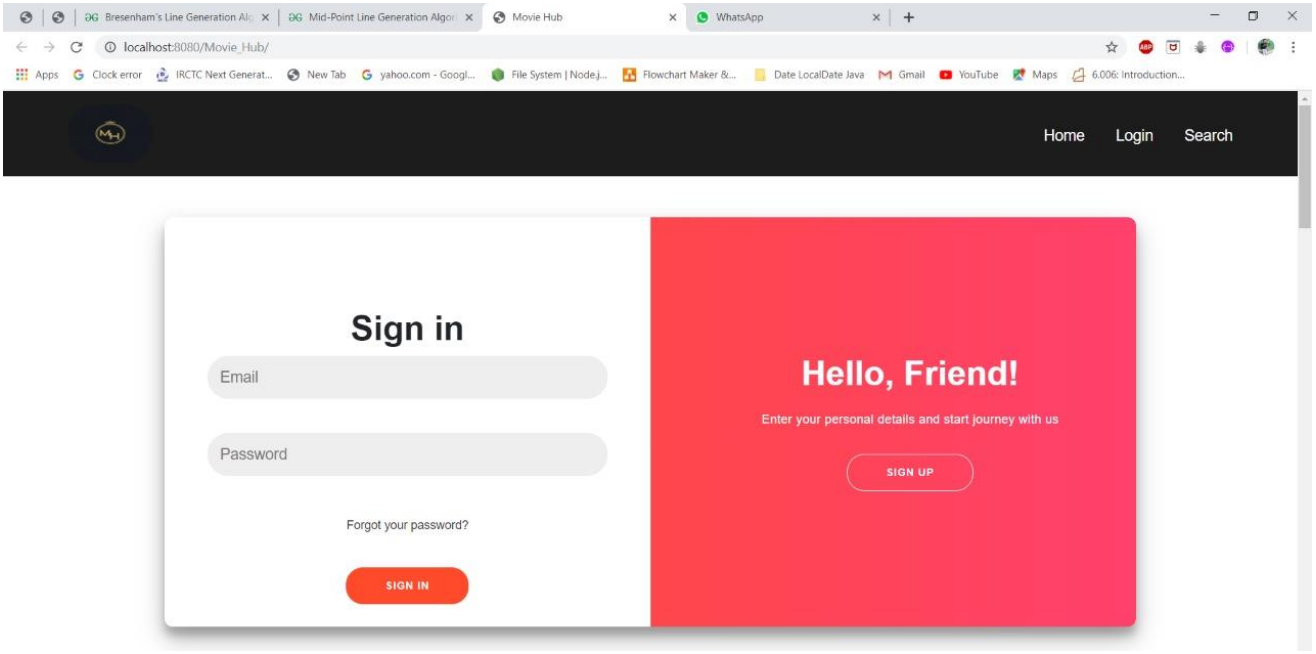
Transactions implementation:

Explain how you will implement the ACID (atomicity, consistency, isolation and durability) properties of transactions (programs that access databases.)

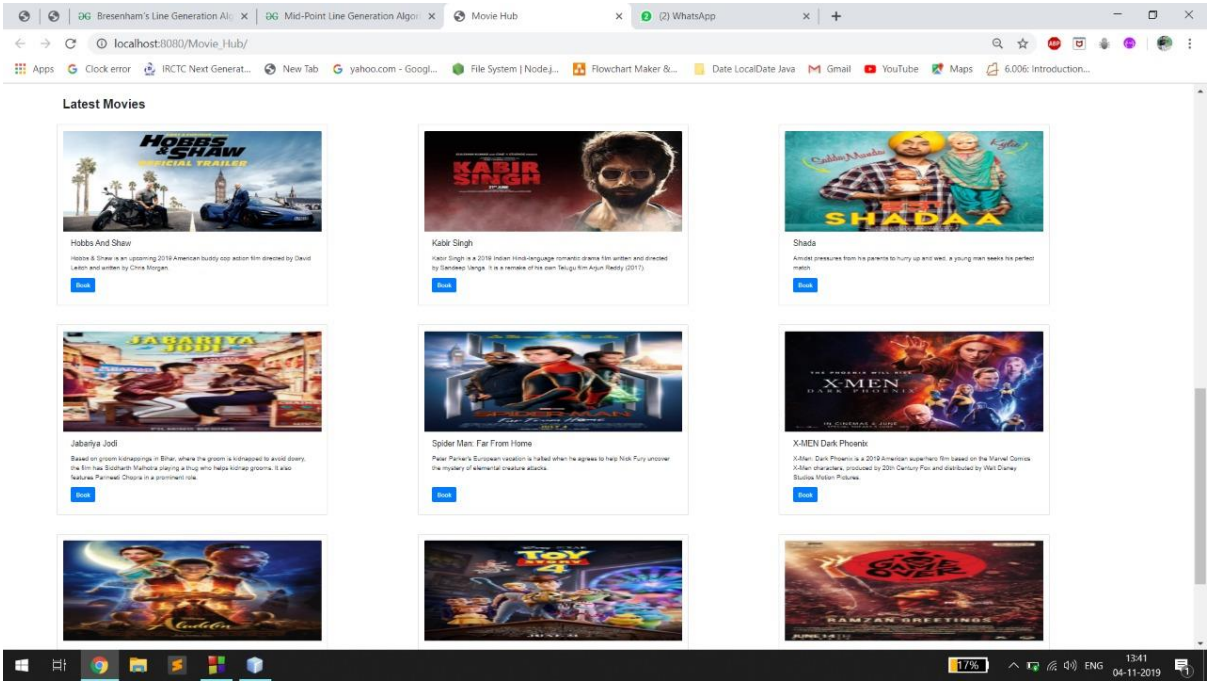
Graphical User Interface

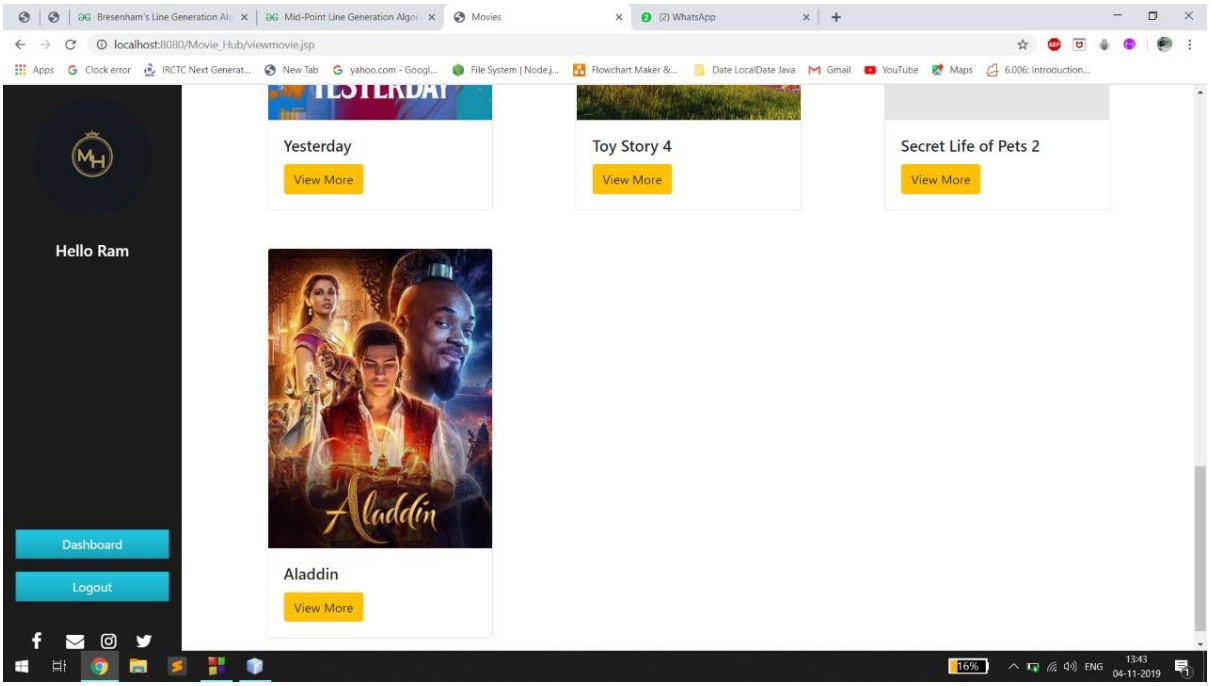
- Provide, in an organized way, the pictures of all the forms in the graphical user interface with a reference to the functional requirement it implements. You may use html to present the graphical user interfaces.
- For each form in the graphical user interface, provide:
    - The names of the controls and fields on that form,
    - The names of the events, methods, or procedures that cause that form to be displayed, and
    - The names of the events, methods, or procedures triggered by each control.

Home Page


Movie Hub
WhatsApp
localhost:8080/Movie_Hub/
Home
Login
Search
Sign in
Email
Password
Forgot your password?
SIGN IN
Hello, Friend!
Enter your personal details and start journey with us
SIGN UP


FAST & FURIOUS PRESENTS
HOBBS & SHAW
OFFICIAL TRAILER
LF68 LVO


Movie Hub
localhost:8080/Movie_Hub/
Latest Movies
Hobbs And Shaw
Kabir Singh
Shada
Jabariya Jodi
Spider Man: Far From Home
X-MEN Dark Phoenix


Movies
localhost:8080/Movie_Hub/viewmovie.jsp
Hello Ram
Yesterday
View More
Toy Story 4
View More
Secret Life of Pets 2
View More
Aladdin
View More
Dashboard
Logout

8.8 Class Diagrams and Classes

Provide a class diagram and an inheritance tree/diagram.

Each method has to be defined[2]:

1. Method Name
2. Parameters, each documented with its intended use
3. Return Value, suitably documented
4. Informal description of what the procedure does
5. Data structure and tables it accesses
6. Pre-conditions: Assumptions the method can make about the state of the global data structures and database when it starts
7. Validity Checks, Errors, and other Anomalous Situations: Validity checks the method must make about the state of the global data structures, the database, and its parameters, including the actions that must be taken when such a check fails.
8. Post-conditions: The changes the method is supposed to make to the global data structures and database.
9. Called by: The methods or events that call it
10. Calls: The methods it calls and any events it causes.

8.9 Uses/Interactions

A description of this component's collaborations with other components. What other components is this entity used by? What other components does this entity use (this would include any side-effects this entity might have on other parts of the system)? This concerns the method of interaction as well as the interaction itself. Object-oriented designs should include a description of any known or anticipated subclasses, superclasses, and metaclasses.

8.10 Resources

Hardware requirements

Server Side:

- Processor: i3 3.5 GHz or higher
- RAM: 8GB or more
- Hard Drive: 10 GB or more

Client side:

- RAM: 2GB or more

Software Requirements

- **Database/server:**

  Derby, Glassfish

- **Application:**

  Google Chrome, Internet Explorer.

- **Manpower Requirements**

  4 students have completed this project.

8.11 Processing

A description of precisely how this component goes about performing the duties necessary to fulfill its responsibilities. This should encompass a description of any algorithms used; changes of state; relevant time or space complexity; concurrency; methods of creation, initialization, and cleanup; and handling of exceptional conditions.

8.12 Interface/Exports

The set of services (resources, data, types, constants, subroutines, and exceptions) that are provided by this component. The precise definition or declaration of each such element should be present, along with comments or annotations describing the meanings of values, parameters, etc. .... For each service element described, include (or provide a reference) in its discussion a description of its important software component attributes (Classification, Definition, Responsibilities, Constraints, Composition, Uses, Resources, Processing, and Interface).

Much of the information that appears in this section is not necessarily expected to be kept separate from the source code. In fact, much of the information can be gleaned from the source itself (especially if it is adequately commented). This section should not copy or reproduce information that can be easily obtained from reading the source code (this would be an unwanted and unnecessary duplication of effort and would be very difficult to keep up-to-date). It is recommended that most of this information be contained in the source (with appropriate comments for each component, subsystem, module, and subroutine). Hence, it is expected that this section will largely consist of references to or excerpts of annotated diagrams and source code. Any referenced diagrams or source code excerpts should be provided at any design reviews.

8.13 Detailed Subsystem Design

Provide a detailed description of this software component (or a reference to such a description). Complex diagrams showing the details of component structure, behavior, or information/control flow may be included in the subsection devoted to that particular component (although, unless they are very large or complex, some of these diagrams might be more appropriately included in the System Architecture section. The description should cover any applicable software component attributes (some of which may be adequately described solely by a source code declaration or excerpt).

9. Source Code Details

SLOC: Source Lines of Code. Use the free software sloc to calculate the SLOC

S.No	Filename	SLOC
1.	Login.jsp	85
2.	Logout.jsp	23
3.	Customerhome.jsp	123
4.	Customerprofile.jsp	145
5.	Moviehall.jsp	377
6.	MovieDetail.jsp	158
7.	ViewHall.jsp	124
8.	ViewMovie.jsp	170
9.	ViewOffer.jsp	137
10.	Cart.jsp	128
11.	Book.jsp	98
12.	RegUser.jsp	73
13.	CheckMail.jsp	50
14.	CheckOtp.jsp	40
15.	ForgetPassword.html	20
16.	BookHistory.jsp	155
17.	BookSuccess.jsp	84
18.	Index.html	223
19.	Sendmail.jsp	87
20.	ticketGenerator.jsp	68
21.	updateCustomerProfile.jsp	64
22.	updatePassword.jsp	43

Source Code: -

Index.html

```
<!DOCTYPE html>
<!--
To change this license header, choose License Headers in Project Properties.
To change this template file, choose Tools | Templates
and open the template in the editor.
-->
<html>
   <head>
      <title>Movie Hub</title>
      <meta charset="UTF-8">
      <meta name="viewport" content="width=device-width, initial-scale=1.0">
      <link rel="stylesheet" type="text/css" href="CSS/style.css">
      <link href="https://stackpath.bootstrapcdn.com/font-awesome/4.7.0/css/font-awesome.min.css"
  rel="stylesheet" integrity="sha384-
wvfXpqpZZVQGK6TAh5PVlGOfQNHSoD2xbE+QkPxCAFlNEevoEH3Sl0sibVcOQVnN"
crossorigin="anonymous">
      <link href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T"
crossorigin="anonymous">
   </head>
   <body>


         <!-- Navbar     -->
      <header>
                        <div class="logo">
                              <img src="Images/logo.png">
                        </div>
                        <div class="link-container">
                              <br>
```

```
                        <a href="index.html">Home</a>
                        <button id="login-btn">Login</button>
                        <button id="search-btn">Search</button>
                    </div>
  </header>

  <!--      login-->
  <div class="container" id="container">

<!-- Sign Up form code goes here -->
<div class="form-container sign-up-container">
 <form action="reguser.jsp" method="post">
  <h1>Create Account</h1>
  <input type="text" placeholder="Name" name="nm" required="" /><br>
  <input type="email" placeholder="Email" name="em" required="" /><br>
  <input type="password" placeholder="Password" name="pass" required="" /><br>
  <button type="submit">Sign Up</button>
 </form>
</div>
<!-- Sign In form code goes here -->
<div class="form-container sign-in-container">
 <form action="login.jsp" method="post">
  <h1>Sign in</h1>
  <input type="email" name="em" placeholder="Email" required="" /><br>
  <input type="password" name="pass" placeholder="Password" required="" /><br>
  <a href="forgetpassword.html">Forgot your password?</a><br>
  <button type="submit">Sign In</button>
 </form>
</div>

<!-- The overlay code goes here -->
```

```
    <div class="overlay-container">
      <div class="overlay">
        <div class="overlay-panel overlay-left">
          <h1>Welcome Back!</h1>
          <p>
            To keep connected with us please login with your personal info
          </p>
          <button class="ghost" id="signIn">Sign In</button>
        </div>
        <div class="overlay-panel overlay-right">
          <h1>Hello, Friend!</h1>
          <p>Enter your personal details and start journey with us</p>
          <button class="ghost" id="signUp">Sign Up</button>
        </div>
      </div>
    </div>

</div>

      <!--      Search -->
      <section id="search">
    <div class="search-container">
        <div class="search-container-left">
          <h1>Hello There</h1>
          <p>Looking for Something in specific. Find it here in no time.</p>
        </div>
        <div class="search-container-right">
          <h1>Search Here</h1>
          <form action="#" method="get">
            <input type="text" name="search-text" placeholder="Movie or Hall Name" required=""><br>
```

```
            <span>
              <input type="radio" name="search-group" value="movie" checked=""><label>Movie</label>
            </span><br>
            <span>
              <input type="radio" name="search-group" value="hall"> <label> Halls</label>
            </span><br>

            <button type="submit" id="search-btn">Search</button>
          </form>
        </div>
      </div>
    </section>

      <!-- Scrolling Movies-->
      <section id="scroller">
            <div id="carouselExampleSlidesOnly" class="carousel slide" data-ride="carousel">
            <div class="carousel-inner">
                  <div class="carousel-item active">
                        <img class="scrolling-posters" src="Images/hobbs.jpg" class="d-block w-100"
alt="...">
                  </div>
                  <div class="carousel-item">
                        <img class="scrolling-posters" src="Images/xmen.jpg" class="d-block w-100"
alt="...">
                  </div>
                  <div class="carousel-item">
                        <img class="scrolling-posters" src="Images/shada.jpg" class="d-block w-100"
alt="...">
                  </div>
            </div>
            </div>
```

```
    </section>


    <!--    Movie Cards-->
    <section id="movie-cards">
        <h2>Latest Movies</h2>
        <div class="row">
            <div class="card col-sm-12 col-md-6 col-lg-3 mr-auto" style="width: 18rem;">
            <img src="Images/hobbs.jpg" class="card-img-top" alt="...">
            <div class="card-body">
                <h5 class="card-title">Hobbs And Shaw</h5>
                <p class="card-text">Hobbs & Shaw is an upcoming 2019 American buddy cop action film directed by David Leitch and written by Chris Morgan.</p>
                <a href="#" class="btn btn-primary">Book</a>
            </div>
            </div>
        <div class="card col-sm-12 col-md-6 col-lg-3 mr-auto" style="width: 18rem;">
            <img src="Images/kabir.jpg" class="card-img-top" alt="...">
            <div class="card-body">
                <h5 class="card-title">Kabir Singh</h5>
                <p class="card-text">Kabir Singh is a 2019 Indian Hindi-language romantic drama film written and directed by Sandeep Vanga. It is a remake of his own Telugu film Arjun Reddy (2017).</p>
                <a href="#" class="btn btn-primary">Book</a>
            </div>
        </div>
        <div class="card col-sm-12 col-md-6 col-lg-3 mr-auto" style="width: 18rem;">
            <img src="Images/shada.jpg" class="card-img-top" alt="...">
            <div class="card-body">
                <h5 class="card-title">Shada</h5>
                <p class="card-text">Amidst pressures from his parents to hurry up and wed, a young man seeks his perfect match.</p>
                <a href="#" class="btn btn-primary">Book</a>
```

```
            </div>
        </div>
        </div>

        <div class="row">
            <div class="card col-sm-12 col-md-6 col-lg-3 mr-auto" style="width: 18rem;">
            <img src="Images/jabariya.jpg" class="card-img-top" alt="...">
            <div class="card-body">
                <h5 class="card-title">Jabariya Jodi</h5>
                <p class="card-text">Based on groom kidnappings in Bihar, where the groom is kidnapped to avoid dowry, the film has Siddharth Malhotra playing a thug who helps kidnap grooms. It also features Parineeti Chopra in a prominent role.</p>
                <a href="#" class="btn btn-primary">Book</a>
            </div>
            </div>
        <div class="card col-sm-12 col-md-6 col-lg-3 mr-auto" style="width: 18rem;">
            <img src="Images/spidy.jpg" class="card-img-top" alt="...">
            <div class="card-body">
                <h5 class="card-title">Spider Man: Far From Home</h5>
                <p class="card-text">Peter Parker's European vacation is halted when he agrees to help Nick Fury uncover the mystery of elemental creature attacks.</p>
                <a href="#" class="btn btn-primary">Book</a>
            </div>
        </div>
        <div class="card col-sm-12 col-md-6 col-lg-3 mr-auto" style="width: 18rem;">
            <img src="Images/xmen.jpg" class="card-img-top" alt="...">
            <div class="card-body">
                <h5 class="card-title">X-MEN Dark Phoenix</h5>
                <p class="card-text">X-Men: Dark Phoenix is a 2019 American superhero film based on the Marvel Comics X-Men characters, produced by 20th Century Fox and distributed by Walt Disney Studios Motion Pictures.</p>
                <a href="#" class="btn btn-primary">Book</a>
            </div>
```

```
</div>
</div>

<div class="row">
    <div class="card col-sm-12 col-md-6 col-lg-3 mr-auto" style="width: 18rem;">
    <img src="Images/aladin.jpg" class="card-img-top" alt="...">
    <div class="card-body">
        <h5 class="card-title">Aladdin</h5>
        <p class="card-text">Aladdin is a 2019 American musical fantasy film directed by Guy Ritchie, who co-wrote the screenplay with John August. Produced by Walt Disney Pictures, it is a live action adaptation of Disney's 1992 animated film of the same name, which is based on the eponymous tale from One Thousand and One Nights.</p>
        <a href="#" class="btn btn-primary">Book</a>
    </div>
    </div>
<div class="card col-sm-12 col-md-6 col-lg-3 mr-auto" style="width: 18rem;">
    <img src="Images/toy.jpg" class="card-img-top" alt="...">
    <div class="card-body">
        <h5 class="card-title">Toy Story 4</h5>
        <p class="card-text">Toy Story 4 is a 2019 American computer-animated comedy film produced by Pixar Animation Studios for Walt Disney Pictures. It is the fourth installment in Pixar's Toy Story series and the sequel to 2010's Toy Story 3.</p>
        <a href="#" class="btn btn-primary">Book</a>
    </div>
</div>
<div class="card col-sm-12 col-md-6 col-lg-3 mr-auto" style="width: 18rem;">
    <img src="Images/gameover.jpg" class="card-img-top" alt="...">
    <div class="card-body">
        <h5 class="card-title">Game Over</h5>
        <p class="card-text">Game Over is a 2019 Indian Tamil-Telugu bilingual thriller film written by Ashwin Saravanan and Kaavya Ramkumar and directed by Ashwin Saravanan.</p>
        <a href="#" class="btn btn-primary">Book</a>
    </div>
```

```
            </div>
            </div>


        </section>


        <footer>
            <p>&copy Copyright 2019</p>
            <div>
                <a href="#"><i class="fa fa-facebook fa-lg social-icons"></i></a>
                <a href="#"><i class="fa fa-envelope fa-lg social-icons"></i></a>
                <a href="#"><i class="fa fa-instagram fa-lg social-icons"></i></a>
                <a href="#"><i class="fa fa-twitter fa-lg social-icons"></i></a>
            </div>
        </footer>


        <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-
q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
crossorigin="anonymous"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"
integrity="sha384-UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1"
crossorigin="anonymous"></script>
    <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"
integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM"
crossorigin="anonymous"></script>



    <script type="text/javascript" src="Scripts/main.js"></script>
    </body>
</html>
```

CustomeHome.jsp

```
<%--
   Document   : customerhome
   Created on : 2 Jul, 2019, 9:30:48 AM
   Author     : VIKAS
--%>


<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
   <head>
      <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
      <title>Home</title>
      <link href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T"
crossorigin="anonymous">
          <link href="https://stackpath.bootstrapcdn.com/font-awesome/4.7.0/css/font-awesome.min.css"
  rel="stylesheet" integrity="sha384-
wvfXpqpZZVQGK6TAh5PVlGOfQNHSoD2xbE+QkPxCAFlNEevoEH3Sl0sibVcOQVnN"
crossorigin="anonymous">


      <link rel="stylesheet" type="text/css" href="CSS/customerhome.css">
      <%@page import="java.sql.*"%>
   </head>
   <body>
      <div class="main-container">
         <div class="sidebar-left">
            <img class="sidebar-logo" src="Images/logo.png">
            <%
               String em = (String)session.getAttribute("email");
               try
         {
```

```
        Class.forName("org.apache.derby.jdbc.ClientDriver");
        Connection cn= DriverManager.getConnection("jdbc:derby://localhost:1527/MovieHubDb");

        PreparedStatement ps= cn.prepareStatement("select * from REGUSER where EMAIL=?");
        ps.setString(1, em);


        ResultSet rs=ps.executeQuery();

        if(rs.next())
        {
           String user_name = rs.getString("name");
           %><h5>Hello <%
           out.print(user_name);
                %></h5><%
        cn.close();
        }

    }
    catch(Exception ex)
    {
        out.print(ex);
    }

            %>

           <h4></h4>

                        <a href="logout.jsp" type="button" class="btn btn-info btn-block"
id="logout">Logout</a>
```

```
                <div>
                        <a href="#"><i class="fa fa-facebook social-icons"></i></a>
                        <a href="#"><i class="fa fa-envelope social-icons"></i></a>
                        <a href="#"><i class="fa fa-instagram social-icons"></i></a>
                        <a href="#"><i class="fa fa-twitter social-icons"></i></a>
                </div>

</div>
        <div class="sidebar-right">
                <a href="customerProfile.jsp">
                        <div>
                        <img src="Images/user.png" width="120" height="120">
                        <h3>Profile</h3>
                        </div>
                </a>
                <a href="viewmovie.jsp">
                        <div>
                                <img src="Images/movie3.webp" width="140" height="140">
                                <h3 class="collection" style="left: -3px;">Movies</h3>
                        </div>
                </a>
                <a href="viewhall.jsp">
                        <div>
                                <img src="Images/hall2.png" width="120" height="120">
                                <h3>Halls</h3>
                        </div>
                </a>
                <a href="viewoffer.jsp">
                        <div>
                                <img src="Images/offer8.png" width="120" height="120">
                                <h3>Offers</h3>
```

```
                    </div>
                </a>
                <a href="bookHistory.jsp">
                    <div>
                        <img src="Images/history2.png" width="120" height="120">
                        <h3>Booking History</h3>
                    </div>
                </a>
                <a href="#">
                    <div>

                    </div>
                </a>
                <a href="#">
                    <div>

                    </div>
                </a>
                <a href="#">
                    <div>

                    </div>
                </a>
            </div>
        </div>
    </body>
</html>
```

BookHistory.jsp

```
<%--
Document   : bookHistory
Created on : 19 Nov, 2019, 4:48:45 AM
Author     : VIKAS
--%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
   <head>
      <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
      <title>History</title>
      <link href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T"
crossorigin="anonymous">
      <link href="https://stackpath.bootstrapcdn.com/font-awesome/4.7.0/css/font-awesome.min.css"
         rel="stylesheet" integrity="sha384-
wvfXpqpZZVQGK6TAh5PVlGOfQNHSoD2xbE+QkPxCAFlNEevoEH3Sl0sibVcOQVnN"
crossorigin="anonymous">
         <link href="https://fonts.googleapis.com/css?family=Montserrat&display=swap" rel="stylesheet">
         <link rel="stylesheet" type="text/css" href="CSS/viewmovie.css">
         <%@page import="java.sql.*"%>

         <style type="text/css">
            .one-line{
               display: inline-block;
               font-size: 20px;
               margin-right: 40px;
               width: 11%;
               margin-bottom: 10px;
            }
         </style>
```

```
</head>
<body>
   <div class="main-container">
      <div class="sidebar-left" style="position: fixed;">
         <img class="sidebar-logo" src="Images/logo.png">
         <%
         String em = (String)session.getAttribute("email");
         try
         {

         Class.forName("org.apache.derby.jdbc.ClientDriver");
         Connection cn= DriverManager.getConnection("jdbc:derby://localhost:1527/MovieHubDb");
         PreparedStatement ps= cn.prepareStatement("select * from REGUSER where EMAIL=?");
         ps.setString(1, em);



         ResultSet rs=ps.executeQuery();

         if(rs.next())
         {
         String user_name = rs.getString("name");
         %><h5>Hello <%
         out.print(user_name);
         %></h5><%
         cn.close();
         }

         }
         catch(Exception ex)
         {
         out.print(ex);
```

```
            }


            %>
            <h4></h4>
            <a href="customerhome.jsp" type="button" class="btn btn-info btn-block mb-2"
id="logout">Dashboard</a>
            <a href="logout.jsp" type="button" class="btn btn-info btn-block">Logout</a>
            <div>
              <a href="#"><i class="fa fa-facebook social-icons"></i></a>
              <a href="#"><i class="fa fa-envelope social-icons"></i></a>
              <a href="#"><i class="fa fa-instagram social-icons"></i></a>
              <a href="#"><i class="fa fa-twitter social-icons"></i></a>
            </div>

          </div>
          <div class="sidebar-right" style="margin-left: 14rem; padding-left: 8rem;">
            <h1 style="color: #E4E9FD;">History</h1>

            <strong>
              <div>
                <div class="one-line">Ticket Id</div>
                <div class="one-line">Movie Name</div>
                <div class="one-line">Hall Name</div>
                <div class="one-line">Show Date</div>
                <div class="one-line">Show Time</div>
                <div class="one-line">No of Seats</div>
              </div>
            </strong>

                        <%
```

```
        try
{

    Class.forName("org.apache.derby.jdbc.ClientDriver");
    Connection cn= DriverManager.getConnection("jdbc:derby://localhost:1527/MovieHubDb");

    PreparedStatement ps= cn.prepareStatement("select * from TICKETS where EMAIL=?");

    ps.setString(1, em);

    ResultSet rs=ps.executeQuery();
    while(rs.next())
    {
        int ticketId = rs.getInt("TICKETID");
        int seats = rs.getInt("SEATS");
        int showId = rs.getInt("SHOWID");

        ps= cn.prepareStatement("select * from shows where showid=?");
        ps.setInt(1, showId);
        ResultSet rs2 = ps.executeQuery();
        rs2.next();
        int movieid = rs2.getInt("movieid");
        int hallid = rs2.getInt("hallid");
        Date showdate = rs2.getDate("show_date");
        Time startTime = rs2.getTime("start_time");

        ps= cn.prepareStatement("select * from hall where hall_id=?");
         ps.setInt(1,hallid);
         rs2=ps.executeQuery();
         rs2.next();
         String hallname = rs2.getString("name");
```

```
            ps= cn.prepareStatement("select * from movies where id=?");
            ps.setInt(1, movieid);
            rs2=ps.executeQuery();
            rs2.next();
            String moviename = rs2.getString("name");

           %>
              <div>
                 <div class="one-line"><%=ticketId%></div>
                 <div class="one-line"><%=moviename%></div>
                 <div class="one-line"><%=hallname%></div>
                 <div class="one-line"><%=showdate%></div>
                 <div class="one-line"><%=startTime%></div>
                 <div class="one-line"><%=seats%></div>
              </div>
           <%
        }

        cn.close();

    }
    catch(Exception ex)
    {
        out.print(ex);
    }

%>


        </div>
```

```
        </div>


    </body>
</html>
```

CART.JSP

```
<%--
    Document   : cart
    Created on : 1 Aug, 2019, 2:49:36 PM
    Author     : VIKAS
--%>



<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Movies</title>
        <link href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T"
crossorigin="anonymous">
          <link href="https://stackpath.bootstrapcdn.com/font-awesome/4.7.0/css/font-awesome.min.css"
  rel="stylesheet" integrity="sha384-
wvfXpqpZZVQGK6TAh5PVlGOfQNHSoD2xbE+QkPxCAFlNEevoEH3Sl0sibVcOQVnN"
crossorigin="anonymous">
        <link href="https://fonts.googleapis.com/css?family=Montserrat&display=swap" rel="stylesheet">
        <link rel="stylesheet" type="text/css" href="CSS/viewmovie.css">
        <%@page import="java.sql.*"%>
```

```
    <style>
        table{
            margin-top: 100px;
            font-size: 20px;
            font-family: 'Montserrat';
        }
        tr:nth-child(odd){
            text-transform: uppercase;
        }
        td{
            padding: 7px 180px;
        }

    </style>
</head>
<body>
    <div class="main-container">
        <div class="sidebar-left" style="position: fixed;">
            <img class="sidebar-logo" src="Images/logo.png">

            <%
                String em = (String)session.getAttribute("email");
                try
        {

            Class.forName("org.apache.derby.jdbc.ClientDriver");
            Connection cn= DriverManager.getConnection("jdbc:derby://localhost:1527/MovieHubDb");

            PreparedStatement ps= cn.prepareStatement("select * from REGUSER where EMAIL=?");
            ps.setString(1, em);
```

```
            ResultSet rs=ps.executeQuery();


            if(rs.next())
            {
               String user_name = rs.getString("name");
               %><h5>Hello <%
               out.print(user_name);
                     %></h5><%
            cn.close();
            }


         }
         catch(Exception ex)
         {
             out.print(ex);
         }


                  %>
               <h4></h4>
               <a href="customerhome.jsp" type="button" class="btn btn-info btn-block mb-2"
id="logout">Dashboard</a>


                           <a href="logout.jsp" type="button" class="btn btn-info btn-block">Logout</a>
                           <div>
                                    <a href="#"><i class="fa fa-facebook social-icons"></i></a>
                                    <a href="#"><i class="fa fa-envelope social-icons"></i></a>
                                    <a href="#"><i class="fa fa-instagram social-icons"></i></a>
                                    <a href="#"><i class="fa fa-twitter social-icons"></i></a>
                           </div>
```

```
</div>
        <div class="sidebar-right" style="margin-left: 14rem; padding-left: 8rem;">
            <h1 style="color: #E4E9FD;">Order Summary</h1>
        <%


        try
{

    Class.forName("org.apache.derby.jdbc.ClientDriver");
    Connection cn= DriverManager.getConnection("jdbc:derby://localhost:1527/MovieHubDb");

    PreparedStatement ps= cn.prepareStatement("select * from shows where showid=?");

    int showdId = Integer.parseInt(request.getParameter("showId"));

    ps.setInt(1, showdId);

    ResultSet rs=ps.executeQuery();

    if(rs.next())
    {
        int movieid = rs.getInt("movieid");
        int hallid = rs.getInt("hallid");
        int showid = rs.getInt("showid");
        int availableSeats = rs.getInt("seats");
        int selectedSeats = Integer.parseInt(request.getParameter("numofseats"));
        Date showdate = rs.getDate("show_date");
        Time startTime = rs.getTime("start_time");
        int price = rs.getInt("price");
        int amount = price * selectedSeats;
```

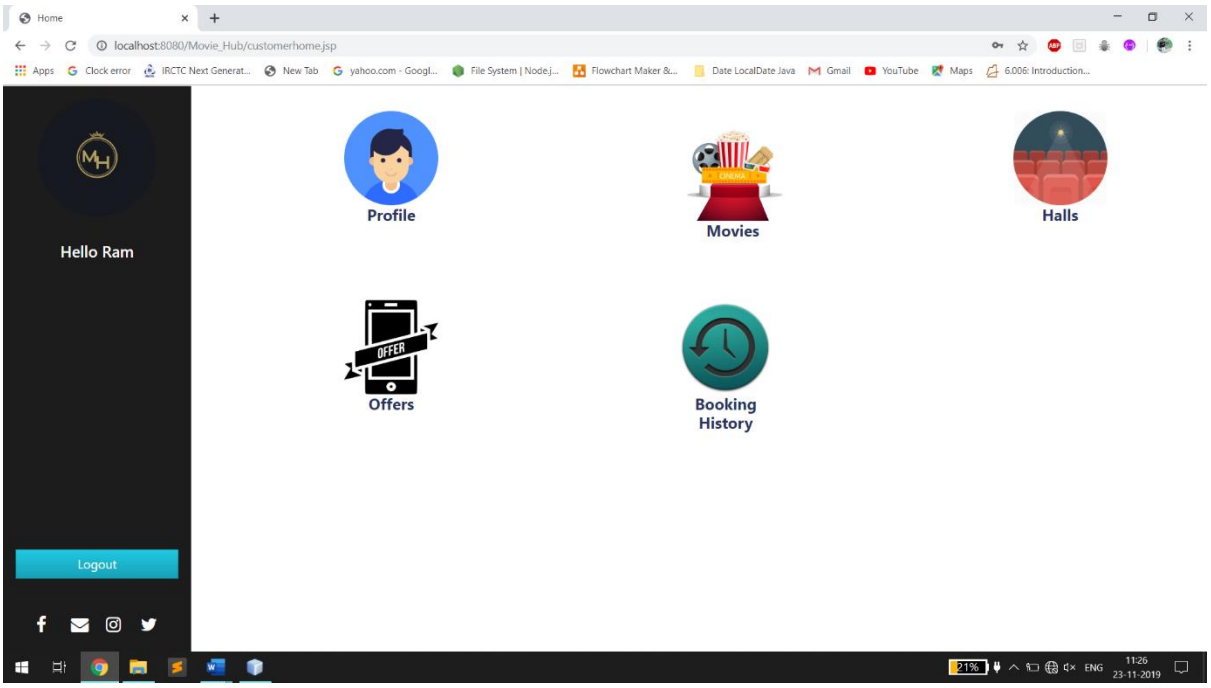
```
//to check if there are seats or not
if(selectedSeats > availableSeats)
{
   %>
   <script>
      alert('No of Seats Available ' + <%=availableSeats%>);
      setTimeout(function(){
         window.location.href = "moviehall.jsp?movieId=<%=movieid%>";
      }, 1);
   </script>




   <%
}



 ps= cn.prepareStatement("select * from hall where hall_id=?");
 ps.setInt(1,hallid);
 rs=ps.executeQuery();
 rs.next();
 String hallname = rs.getString("name");

 ps= cn.prepareStatement("select * from movies where id=?");
 ps.setInt(1, movieid);
 rs=ps.executeQuery();
 rs.next();
 String moviename = rs.getString("name");
```

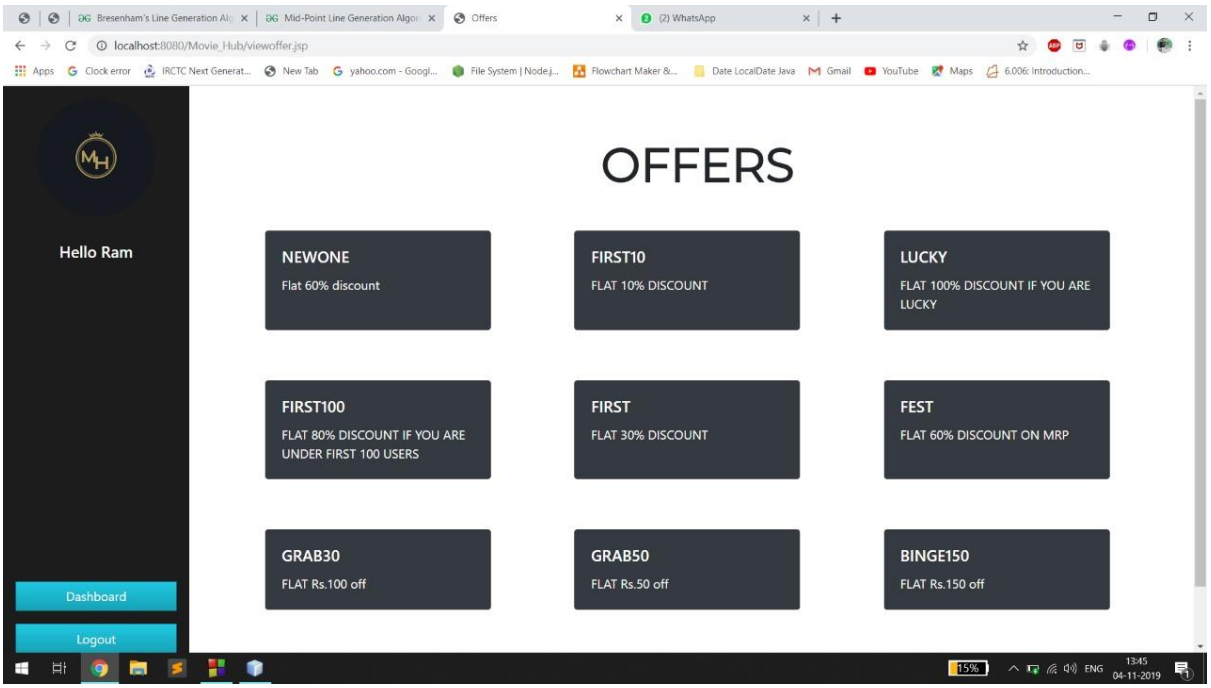
```
%>

<table>
  <tr>
    <td><strong>Movie</strong></td>
    <td><%=moviename%></td>
  </tr>

  <tr>
    <td><strong>Cinema Hall</strong></td>
    <td><%=hallname%></td>
  </tr>

  <tr>
    <td><strong>Seats</strong></td>
    <td><%=selectedSeats%></td>
  </tr>

  <tr>
    <td><strong>Show Date</strong></td>
    <td><%=showdate%></td>
  </tr>

  <tr>
    <td><strong>Show Time</strong></td>
    <td><%=startTime%></td>
  </tr>

  <tr>
    <td><strong>Price of Ticket</strong></td>
    <td><%=price%></td>
```

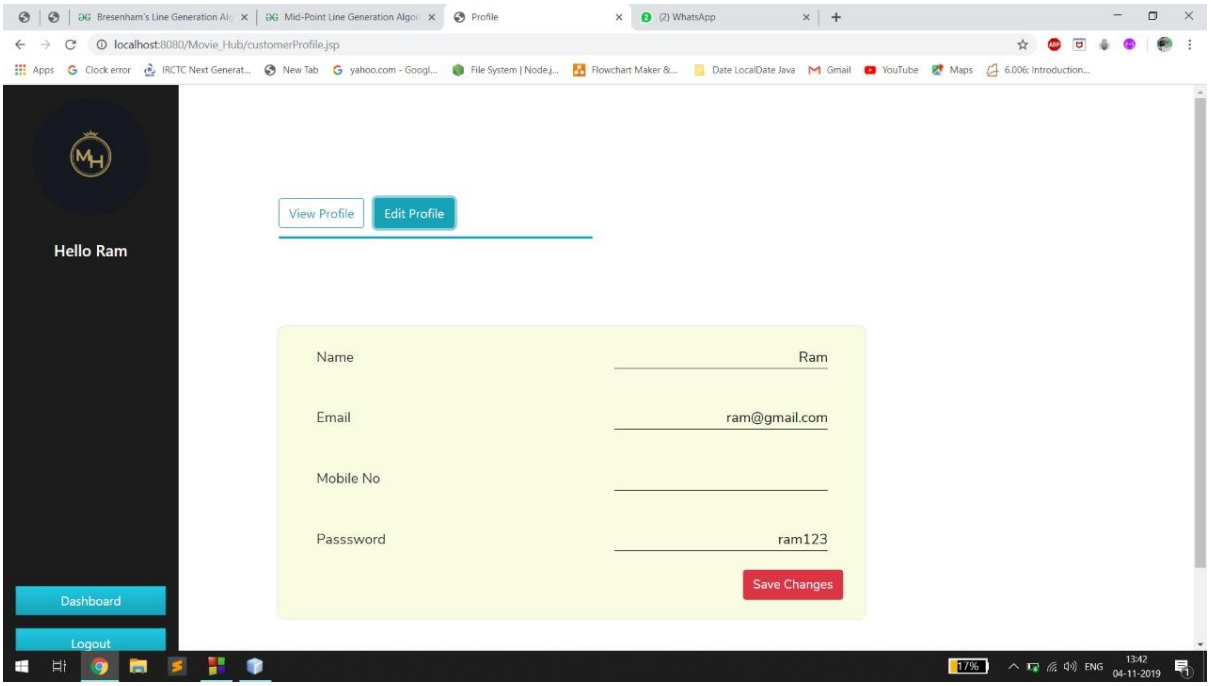
```
                </tr>

                <tr>
                    <td><strong>Amount to Pay</strong></td>
                    <td><%=amount%></td>
                </tr>
              </table>

              <%



             %>

             <a class="btn btn-warning"
href="book.jsp?showId=<%=showid%>&selectedSeats=<%=selectedSeats%>">Proceed to Pay</a>

             <%
            }
            cn.close();

        }
        catch(Exception ex)
        {
            out.print(ex);
        }

              %>



              </div>
```

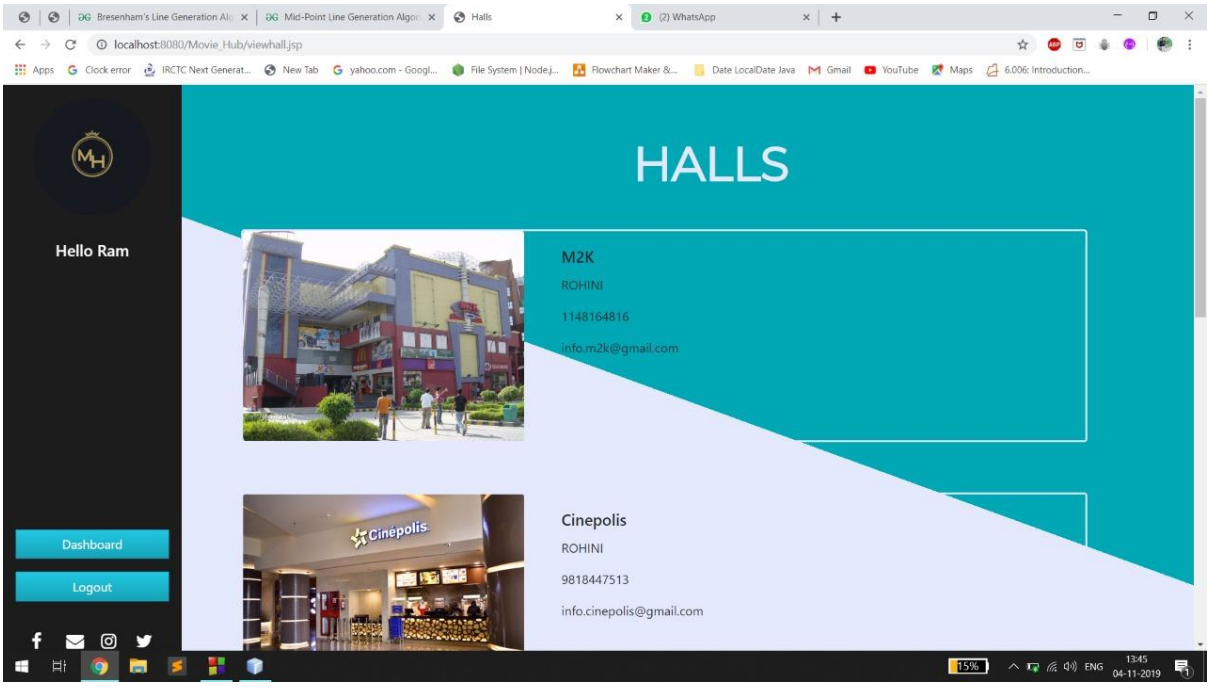
```
        </div>



    </body>
</html>
```

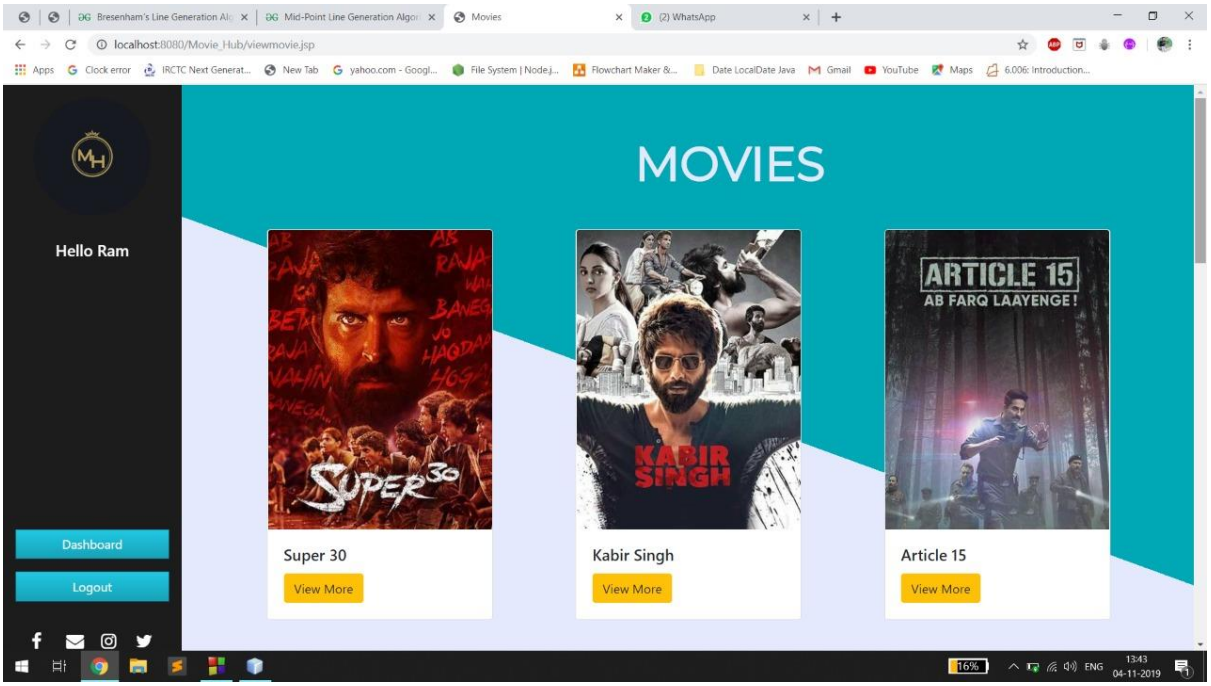
10. Output

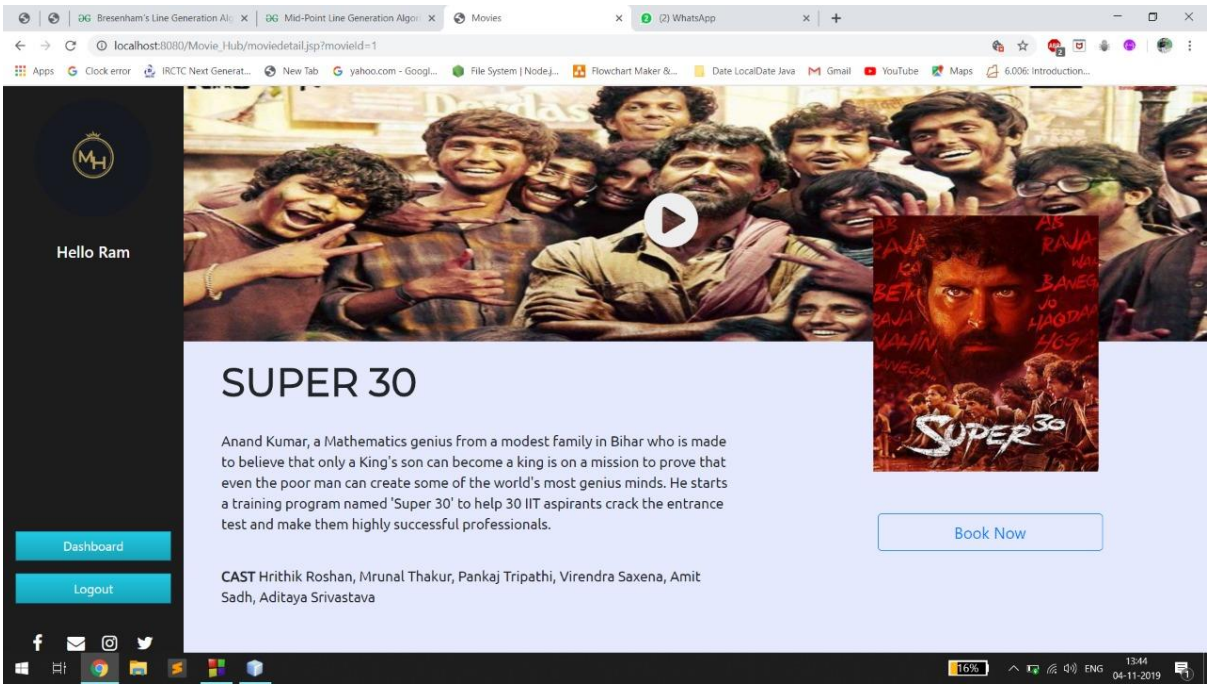

Home
localhost:8080/Movie_Hub/customerhome.jsp
Hello Ram
Profile
Movies
Halls
Offers
Booking History
Logout

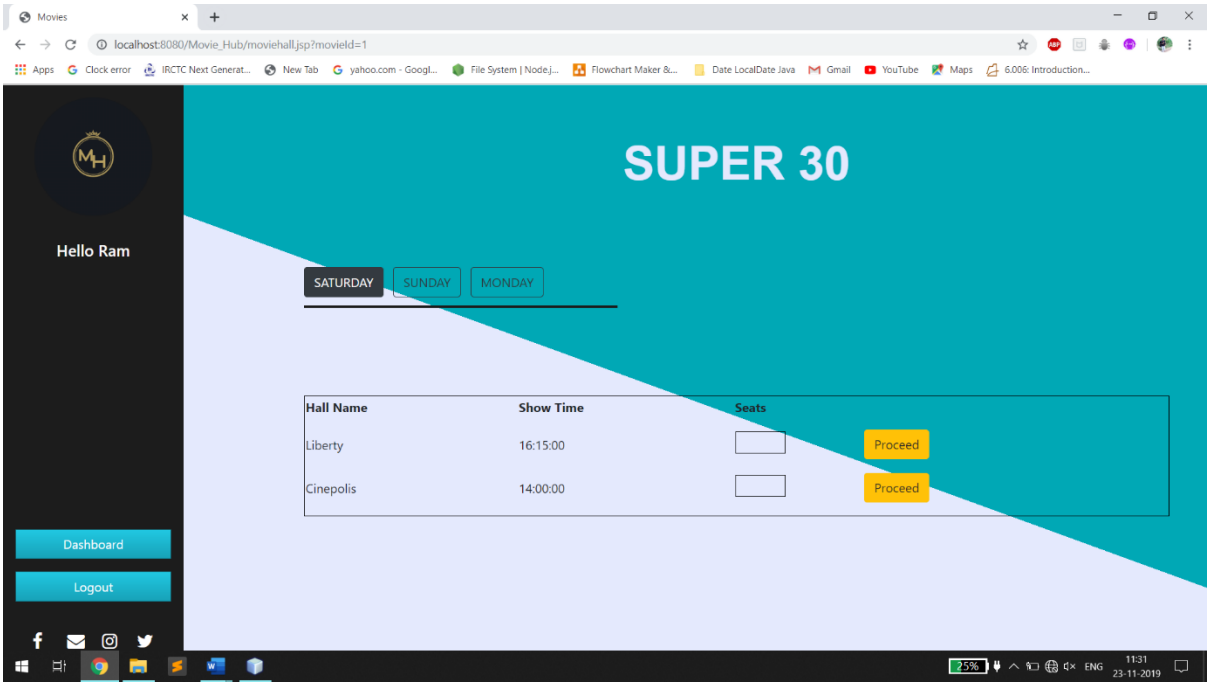

Offers
localhost:8080/Movie_Hub/viewoffer.jsp
Hello Ram
OFFERS
NEWONE
Flat 60% discount
FIRST10
FLAT 10% DISCOUNT
LUCKY
FLAT 100% DISCOUNT IF YOU ARE LUCKY
FIRST100
FLAT 80% DISCOUNT IF YOU ARE UNDER FIRST 100 USERS
FIRST
FLAT 30% DISCOUNT
FEST
FLAT 60% DISCOUNT ON MRP
GRAB30
FLAT Rs.100 off
GRAB50
FLAT Rs.50 off
BINGE150
FLAT Rs.150 off
Dashboard
Logout

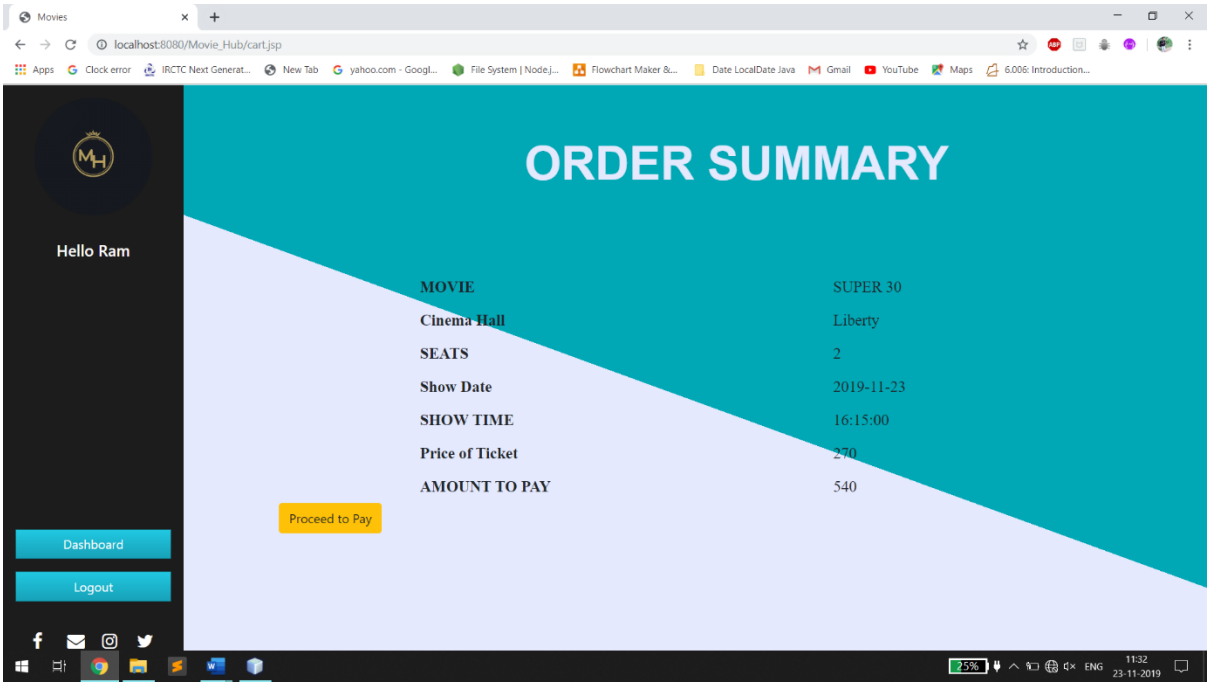

Profile
localhost:8080/Movie_Hub/customerProfile.jsp
Hello Ram
View Profile
Edit Profile
Name
Ram
Email
ram@gmail.com
Mobile No
Passsword
ram123
Save Changes
Dashboard
Logout

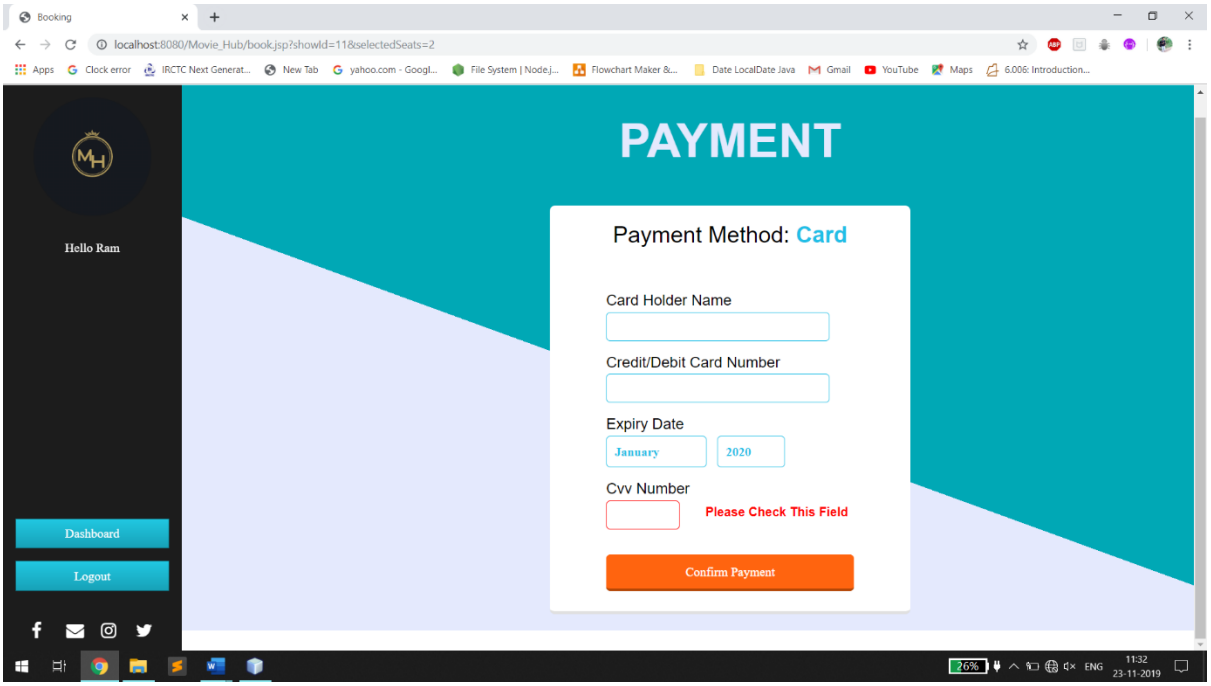

Halls
localhost:8080/Movie_Hub/viewhall.jsp
Hello Ram
HALLS
M2K
ROHINI
1148164816
info.m2k@gmail.com
Cinepolis
ROHINI
9818447513
info.cinepolis@gmail.com
Dashboard
Logout

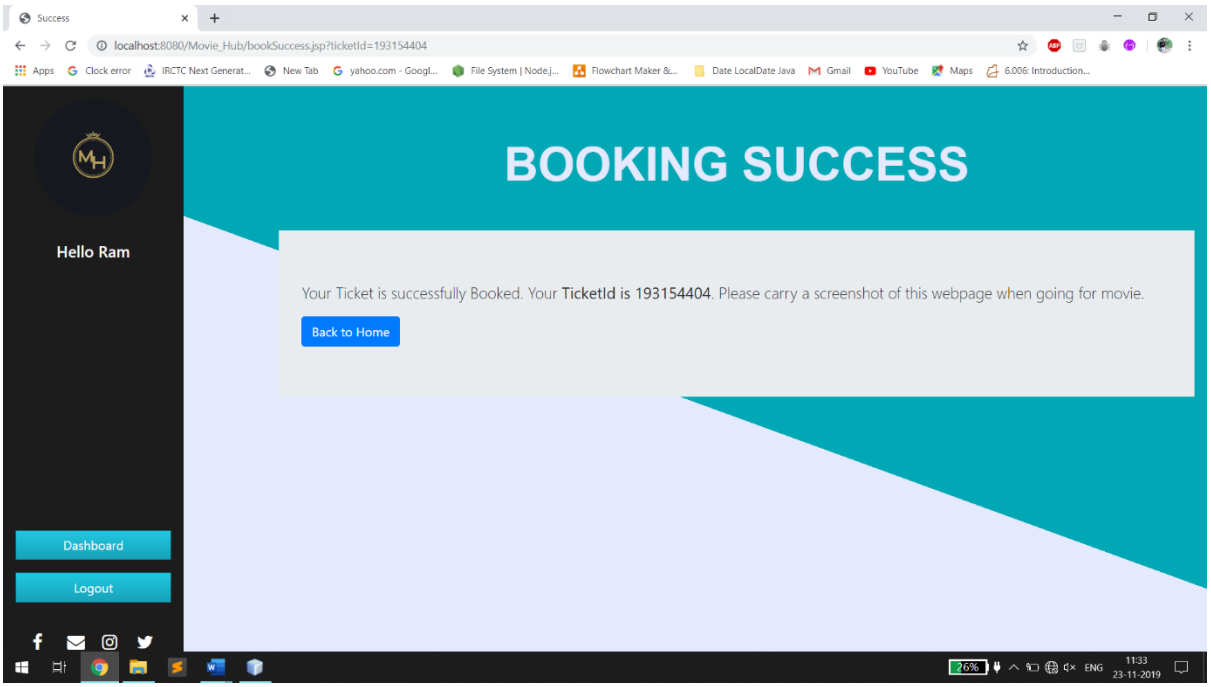

Hello Ram
MOVIES
Super 30
View More
Kabir Singh
View More
Article 15
View More
Dashboard
Logout

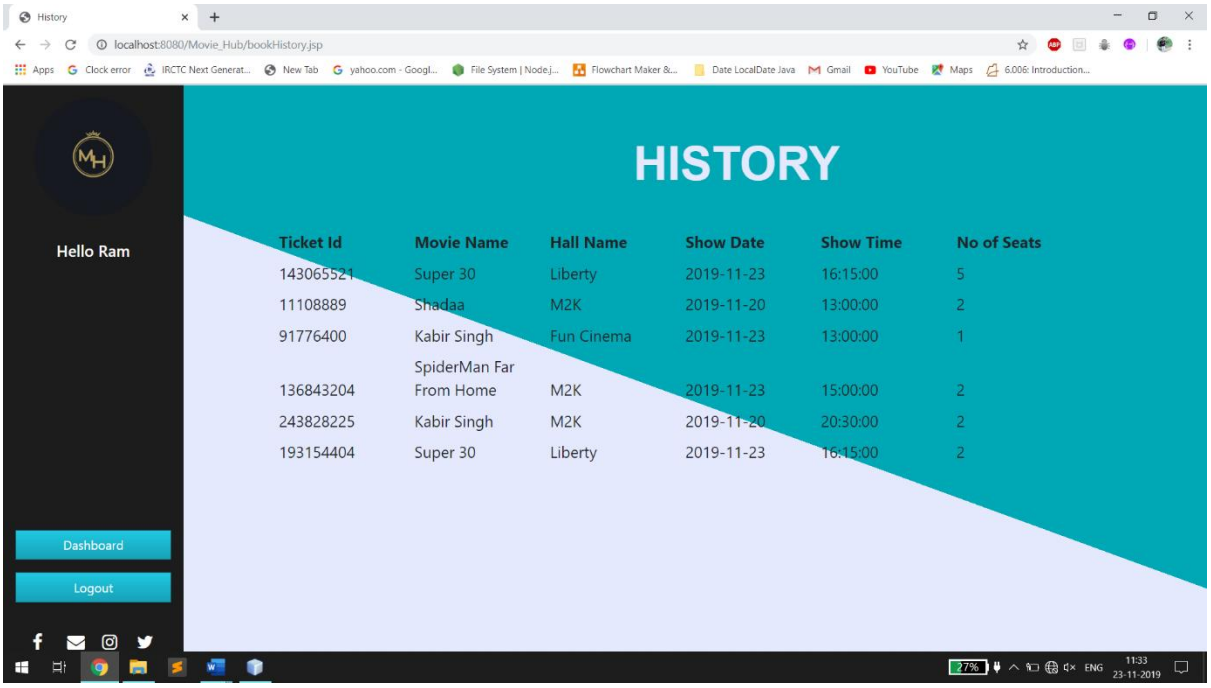

Hello Ram
SUPER 30
Anand Kumar, a Mathematics genius from a modest family in Bihar who is made to believe that only a King's son can become a king is on a mission to prove that even the poor man can create some of the world's most genius minds. He starts a training program named 'Super 30' to help 30 IIT aspirants crack the entrance test and make them highly successful professionals.
CAST Hrithik Roshan, Mrunal Thakur, Pankaj Tripathi, Virendra Saxena, Amit Sadh, Aditaya Srivastava
Book Now
Dashboard
Logout


Movies
localhost:8080/Movie_Hub/moviehall.jsp?movieId=1
Hello Ram
SUPER 30
SATURDAY
SUNDAY
MONDAY
Hall Name
Show Time
Seats
Liberty
16:15:00
Proceed
Cinepolis
14:00:00
Proceed
Dashboard
Logout


Movies
localhost:8080/Movie_Hub/cart.jsp
Hello Ram
ORDER SUMMARY
MOVIE
SUPER 30
Cinema Hall
Liberty
SEATS
2
Show Date
2019-11-23
SHOW TIME
16:15:00
Price of Ticket
270
AMOUNT TO PAY
540
Proceed to Pay
Dashboard
Logout


Booking
localhost:8080/Movie_Hub/book.jsp?showId=11&selectedSeats=2
PAYMENT
Hello Ram
Payment Method: Card
Card Holder Name
Credit/Debit Card Number
Expiry Date
January
2020
Cvv Number
Please Check This Field
Confirm Payment
Dashboard
Logout


Success
localhost:8080/Movie_Hub/bookSuccess.jsp?ticketId=193154404
BOOKING SUCCESS
Hello Ram
Your Ticket is successfully Booked. Your TicketId is 193154404. Please carry a screenshot of this webpage when going for movie.
Back to Home
Dashboard
Logout



11. Glossary

An ordered list of defined terms and concepts used throughout the document.

12. Bibliography

A list of referenced and/or related publications.

Web purchases products from your Internet site, the purchase information is then sent to the Web Services, which totals all the products, adds a record to the accounts receivable database, and then returns a response with an order confirmation number. Not only can this Web Service interact with Web pages, it can interact with other Web Services, such as a corporate accounts payable system.

In order for the Web Service model to survive the natural evolution of programming languages, it must include much more than a simple interface to the Web. The Web service model also includes protocols that enable applications to find Web Services available across a LAN or the Internet. This protocol also enables the application to explore the Web Service and determine how to communicate with it, as well as how to exchange information. To enable Web Service discovery, the Universal Discovery, Description and Integration (UDDI) was established. This allows Web Services to be registered and searched, based on key information such as company name, type of service, and geographic location.

CHAPTER – 4

TESTING AND IMPLEMENTATION

TEST CASES

Sign Up Phase

Date: 20/07/2019 **System:** Movie Hub **Objective:** To test the Sign up Phase **Function:** **Version\Release: Status:** To be Tested			**Tested by:** Vikas Kumar **Environment:** Net beans using Chrome **Test Id:** T_01 **Req. Id:** **Screen:** Browser (chrome) **Test Type:** Manual Testing			
Step Sr.	**Step Description**	**Path & Action**	**Test Data**	**Expected Results**	**Actual Result Pass\Fail**	**Comments**
1	User Sign Up- Verify that blank username is not allowed for user registration	1. Go to Login/Sign Up Page 2. Click on Register me button without putting any name in Name field	Name: Email: abc@test.com Password:abc@123	Application should throw an error message that Blank Name is not allowed.	Pass	
2	User Sign Up- Verify that blank Email is not allowed	1. Go to Login/Sign U p Page 2. Click on	Name: Arunita jaiswal Email:	Application should throw an error	Pass	

	for user registration	Register button without putting any Email in Email field	Password: abc@123	message that Blank Email is not allowed.		
4	User Sign Up- Verify that blank Password is not allowed for user registration	1. Go to Login/Sign U p Page 2. Click on Register me button without Putting any password in the password field	Name: Arunita Jaiswal Email:abc@test.com Password:	Applicatio n should throw an error message that Blank Password is not allowed.	Pass	
5	User Sign Up- Verify that only more than 6 char passowrd is considered as valid password	1. Go to Login/Sign U p Page 2. Click on Register me button with Password less than 6 Character	Name: Arunita Jaiswal	Applicatio n should throw an error message that Your password should be more than 6 letters.	Fail	

			Email:arunitajaiswal@gmail.com Password: aru			
6	User Sign Up- Verify If all the fields are filled properly and password entered is more than 6 char, a user account should be created with username as Email and Password as entered by User.	1. Go to Login/Sign U p Page 2. Click on Register me button	Name: Arunita Jaiswal Phone: 9555335776 Email:arunitajaiswal@gmail.com Password:aru@123	Application should create a new account for user with email as username and password as entered by user.	Pass	
End						

Login/Forget Password Phase

Date: 20/07/2019 **System: Movie Hub Objective:** To test the Login Phase **Function:** **Version\Release: Status:** To be Tested			**Tested by:** Gaurav Singh Thapa **Environment:** Net beans using Chrome **Test Id:** T_02 **Req. Id:** **Screen:** Browser (chrome) **Test Type:** Manual Testing			
Step Sr.	**Step Description**	**Path & Action**	**Test Data**	**Expected Results**	**Actual Result Pass\Fail**	**Comments**
1	Login- Verify that blank username is not allowed for user registeration	1. Go to Login/SignUp Page 2. Click on Login button without putting any name in Name field	Username: Password: aru@123	Application should throw an error message that Blank Name is not allowed.	Pass	
2	Login- Verify that blank Password is not allowed for user registeration	1. Go to Login/SignUp Page 2. Click on Login button without putting any Password in Password field	Username: 123401 Password:	Application should throw an error message that Blank Password is not allowed.	Pass	

3	Login- Verify that user is not logged in with invalid username or password	1. Go to Login/SignUp Page 2. Click on Login button with Invalid Username\ Pa ssword	Username: 1 Password:123465	A new page is open with a messag e that please register yourself.	Fail	
4	Login- Verify that user is allowed to login with valid username and password	1. Go to Login/SignUp Page 2. Click on Login button with valid Username\ Pa ssword	Username:ram@gmail.com Password:ram123	The user is logged in with his account and profie page is opened up by default.	Pass	
5	Forget Passowrd- Verify that if user enters an valid registered email then an reset password is send	1. Go to Forget Password Page 2.Click on Get password button with valid Email id.	Email id:ram@gmail.com	Reset passwor d is sent to user's account and user is able to reset his account	Pass	

				password .		
6	Forget Passowrd- Verify that if user enters an Invalid registered email then an reset password link is send	1. Go to Forget Password Page 2.Click on Get password button with valid Email id.	Email id:123401@test.com	Reset password link is not sent to user's account and an error messag e is generate that please enter a valid email id.	Fail	
End						

Movie Details Phase

Date: 20/07/2019 **System:** Movie Hub **Objective:** To test the description of movies. **Function: Version\Release: Status:** To be Tested			**Tested by:** Rishabh Rawat **Environment:** Net beans using Chrome **Test Id:** T_03 **Req. Id: Screen:** Browser (chrome) **Test Type:** Manual Testing			
Step Sr.	**Step Description**	**Path & Action**	**Test Data**	**Expected Results**	**Actual Result Pass\Fail**	**Comments**
1	Verify that the user is getting the description of each movie he clicks	1.login via username & password. 2. click on Movie Icon 3. Click on the movie you like.	Super 30	User Should be able to see the description and poster of the movie he selects.	Pass	
2	Verify that the user is getting the shows for today and upcoming	1.login via username & password.	Super 30	User should be able to see the list of shows mapped	Pass	

	days only and not the previous days	2. click on Movie Icon 3. Select your preferred movie then click book now.		with the approriate halls along with timing.		
End						

Booking Phase

Date: 20072019 **System:** Movie Hub **Objective:** To test the user ticket booking phase. **Function: Version\Release: Status:** To be Tested			**Tested by:** Abhishek Rawat **Environment:** Net beans using Chrome **Test Id:** T_03 **Req. Id: Screen:** Browser (chrome) **Test Type:** Manual Testing			
Step Sr.	**Step Description**	**Path & Action**	**Test Data**	**Expected Results**	**Actual Result Pass\Fail**	**Comments**
1	To Verify that this is the default page opened up after the same procedure in	1.Login via username & password. 2. click on book Ticket 3. Complete Payment by filling card details.	Username:123401 Password:aru@123	A new page should open which should display the ticket id related to users booking.	Pass	

	above test					
End						

Booking History Phase

Date: 20072019 **System:** Movie Hub **Objective:** To test the Booking History Phase **Function: Version\Release:** **Status:** To be Tested			**Tested by:** Rishabh Rawat **Environment:** Net beans using Chrome **Test Id:** T_07 **Req. Id: Screen:** Browser (chrome) **Test Type:** Manual Testing			
Step Sr.	**Step Description**	**Path & Action**	**Test Data**	**Expected Results**	**Actual Result Pass\Fail**	**Comments**
1	To verify that the user accesses all its records regarding booking History.	1.Login via username & password. 2. click on Booking History	None Required Details Directed in Account	User should be able to see all the bookings he has done with that account.	Pass	
End						